Re: Building high available and high performance web sites with MS

Source:
http://www.tech−archive.net/Archive/SQL−Server/microsoft.public.sqlserver.clustering/2006−11/msg00105.html

- *From*: "Anthony Thomas" <ALThomas@xxxxxxxxx>
- *Date*: Sat, 18 Nov 2006 10:58:27 −0600

There are two technologies that you can use in conjunction with each other, if desired, but they are TWO separate technologies, each supporting very different goals.

Failover clustering, a.k.a. server clusters, support redundant hardware that exposes a single virtual host name. In the case of hardware failure, the other node picks up the services, and continues to present the same virtual server to the network. It is a fail−safe, business resumption solution.

Federated servers, on the other hand, is the closest thing you can come to a NLB solution for database services. A collection of servers host separate instances of SQL Server, all with a copy of the database schema; however, the base tables are partitioned vertically, by date, name, or some other partitioning function, with linked servers to the other systems. Each database contains copies of common distributed portioned views, that aggregate the base tables from across the federation.

Now, since each server exposes this same view across the entire federation, each is able to process data from all systems. A naive solution, would be to introduce a load balancing algorithm to distribute client connections equally across this federation. However, to make optimal use of such a solution, one would need the application to code which server to connect to based on the data and which partitioned server supported that data locally, Data Dependent Routing.

Much of the SS2K documentation is still applicable to SS2K5, but I would search them both as there have been some enhancements and improvements.

Failover Clustering
http://msdn2.microsoft.com/en−gb/library/ms189134.aspx

SQL Server 2000 Failover Clustering
http://www.microsoft.com/technet/prodtechnol/sql/2000/maintain/failclus.mspx

Scaling Out SQL Server with Data Dependent Routing
http://www.microsoft.com/technet/prodtechnol/sql/2005/scddrtng.mspx

Federated SQL Server 2000 Servers
http://msdn.microsoft.com/library/default.asp?url=/library/en−us/architec/8_ar_cs_4fw3.asp

Federated Database Servers
http://msdn2.microsoft.com/en−us/library/ms190381.aspx

Best of luck.

Sincerely,

Anthony Thomas

−−

"Flaxen" <Flaxen@xxxxxxxxxxxxxxxxxxxxxxxxx> wrote in message
news:C29A1C86−1103−4983−BDB6−77C2740EDA1C@xxxxxxxxxxxxxxx

> What is the purpose of adding extra nodes past the 2 node cluster if
> clustering is just for failover? I read about adding extra nodes when
> researching and didnt see why that would be advantageous? In case

multiple

> servers malfunction at the same time?
>
> Ok, now that I know Clustering wont solve my problem, is there another
> method of doing database load balancing? Would have lets say 3 servers,

each

> acting as a publisher/subscriber and then a 2−node cluster acting as a
> distributer work well in this situation? I feel in this solution, i would
> have 3 sql servers with the same info and then a high available

distributer.

> Then I assume I would need a way to make the 3 sql servers appear as one
> server somehow so the code only has to reference one sql server. Maybe

round

> robin dns?
>
> Am i making this too difficult? Is there an easier solution? I cant
> imagine I am the only one who needs a solution where you can scale

Microsoft

SQL servers effectively.

Thanks in advance.
"Geoff N. Hiten" wrote:

You are asking about two goals. First let me addres the availability issue.

Microsoft clustering is a failover technology where a SQL instance can move

between host computers in case of hardware or systems failure. The basic

single−instance, two−node cluster (sometimes incorrectly called Active/Passive) uses only one host at a time. You can add additional nodes

(host computers) and instances, but the instances do not share information

and are not "brick−type" scalable.

Right now, the way to scale SQL is to replace the host computer with a larger one. With failover clustering, you can do this with minimal downtime

by replacing one node at a time.

−−
Geoff N. Hiten
Senior Database Administrator
Microsoft SQL Server MVP

"Flaxen" <Flaxen@xxxxxxxxxxxxxxxxxxxxxxxxx> wrote in message news:7A300D7D−7E43−49DF−A3A7−9A134C55AA22@xxxxxxxxxxxxxxx

Hi all. My first time posting to this newsgroup, but I think I am in

the

right place. I am in the discovery stage of building a web site that

we

anticipate will have a large amount of traffic. Our design platform

is

Windows and Microsoft SQL and the goal is to have web sites and

database

servers that are not only high available but also scalable that I can

put

in
a new server as needed to increase performance.

For the web sites, I know that I can use the NLB services of Windows

2003

standard edition. My lack of knowledge, though, is on the MS SQL

side. I

have been reading a lot of info on Microsoft SQL server failover, but

from

what I have read this tyipcally entails having two MS SQL servers

running

in
an active/passive mode.

Does Microsoft SQL Server support the ability to have multiple MS SQL
servers using the same database and appearing as one physically

database?

If
one fails, the other servers pick up the load, and if we need to
add a

new

server we simply install it into the cluster?

I understand that I will need enterprise edition to accomplish
the

goals

above, but are these goals even doable?

I know I can setup a bunch of SQL servers all doing
transaction
replication, but that makes each server appear as its own
database

server

and
I feel is not the proper solution for this.

Hope my questions make sense and looking forward to the
answers.

.