

<http://www.db2mag.com/story/showArticle.jhtml?articleID=201201093>

Informix DBA: IDS 11 Races Ahead

By Lester Knutsen

"Cheetah" is faster and better than previous IDS releases.



A lot has happened in the last month; I started this article on a plane trip to Beijing, China, and finished it after attending the IDS "Cheetah" T3 training at the IBM lab in Toronto, Canada. By the time you read this, IDS 11 (code-named "Cheetah") will have been released. I thought the release of IDS 10 was exciting, but I'm even more impressed by IDS 11.

The "Cheetah" open beta program — the first for any Informix product — involved more than 1,000 users who downloaded and tested the new features. My company participated in the early beta, and I'll share a few reflections on the key features that I've seen.

IDS "CHEETAH" TEST SPEEDS

I asked Thomas Beebe, a DBA who works for me and has done most of our internal testing, for his observations. The next paragraphs recount Tom's testing experience.

The new IDS offering has greatly increased the speed of even typical database uses. For testing, Tom used a backup data set provided by wikimedia. It contained a backup snapshot of the wikipedia.org data in XML format. Using some Perl scripts, he loaded the data into "Cheetah" and a matching 9.4 database, both running 64-bit on Solaris 10. This gave more than four million rows with varying data types, including a large CLOB (character large object) column.

The data load took 436 minutes on "Cheetah," compared to 470 on IDS 9.4. Next, he tested dbexport, which was tweaked for the new release. The functionality was the same, but it took only 166 minutes to export the entire data set to disk on "Cheetah," and 193 minutes on IDS 9.4. Building a static index on a varchar was slightly faster on "Cheetah" (56 minutes as opposed to 60 on IDS 9.4).

The most interesting speed tests involved some simple SQL queries. On a large SUM function, "Cheetah" finished in 1:56, and IDS 9.4 took 2:31. On a simple count retrieval using the LIKE clause, "Cheetah" finished in 1:20 and IDS 9.4 finished in 1:52. Most of the time, the newest release has a significant advantage over previous Informix releases.

Tom's tests were performed on a database with a complex set of data types. For classic OLTP using characters and numbers, the speed increase can be even more dramatic.

THE NEW "CHEETAH" CHECKPOINT

One of the reasons for the performance improvement in loads, inserts, updates, and deletes is the new checkpoint function. A checkpoint, when all dirty or changed pages in memory are written to disk, is one of the most disk I/O intensive operations the database server conducts. In earlier IDS versions (through 7.3), when a sync checkpoint was performed, all other update activity on the server was blocked while that data was written to disk. If you had a large amount of memory for `IDS_BUFFERS`, this process could take a while. One of my systems has 6GB for `BUFFERS`; if 50 percent are dirty, IDS is writing out 3GB of data to disk. Imagine how long it takes to copy a 3GB file — that's how long it takes to write out 3GB of `BUFFERS`.

Related Reading

[Informix's New Conference](#)

[Cisco's Informix DBA Leads the Pack](#)

[Online Exclusive: Purifying XML with IDS](#)

In a sync checkpoint, all users doing updates are blocked until this operation finishes. IDS 9.2x introduced fuzzy checkpoints, which helped performance because they reduced the time user activity was blocked. However, fuzzy checkpoints increased the dependence on the Logical Log for logging operations while writing the dirty pages to disk and could cause longer recovery times.

One of the performance tips we used in IDS 7.x and 9.x was to lower the `LRU_MIN_DIRTY` and `LRU_MAX_DIRTY` parameters in the `ONCONFIG` file. These parameters controlled when background writes would flush data to disk and are expressed as a percent of `BUFFERS`, so `LRU_MAX_DIRTY = 60` would mean to start flushing data when 60 percent of the `BUFFERS` are dirty. By making the numbers lower, fewer dirty `BUFFERS` are in memory; a checkpoint would have less work to perform and would perform faster. However, because the LRU threads had to do more work, more CPU resources were consumed by the server (and this was often unnoticed). In one of my tuning classes, I taught an exercise on tuning these parameters and observing the relationship between checkpoint time, the percent of dirty `BUFFERS`, and CPU usage. In very intensive benchmarks or data warehouse loads, I would often find that you had better performance increasing these parameters, because that reduced the overall CPU usage and provided more optimized writes to disk. Checkpoint writes are also ordered by page location on disk; therefore, they're faster.

In the new release, this has all changed. A new checkpoint function doesn't block user updates the way the old sync checkpoint did, and it logs dirty pages to the physical log, so recover time is improved. Therefore, we can start experimenting with increasing the `LRU_MIN_DIRTY` and `LRU_MAX_DIRTY` parameters in the `ONCONFIG` file in OLTP systems. These changes could mean less CPU usage for LRU cleaning activity and better performance of checkpoint writes to get data to disk.

NEW "CHEETAH" REPLICATION FOR HIGH AVAILABILITY

[Preview: IDS Purifies XML](#)

By Jean Georges Perrin

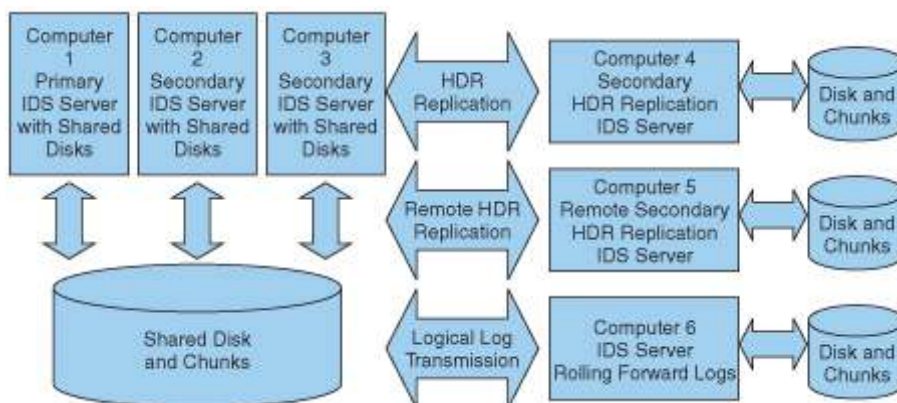
As an application developer, I love Informix Dynamic Server (IDS) 11 (code-named "Cheetah" before its release). Why get so excited? Because "Cheetah" brings tools that make my day-to-day life easier. I also love XML. One can say a lot about XML, but for me, XML is the sign of a new generation of IT professionals. It's the clear separation between a generation focusing on don't-waste-that-extra-byte and a generation focusing on making IT understandable by its peers.

And now, my favorite engine embraces my favorite language. In my article on DB2mag.com, I'll show you how my company, GreenIvory, uses XML and IDS 11 for Web publishing and how these technologies can benefit your company, too. You'll learn what happens when Cheetah speaks XML and how it can work to your benefit.

I find the most exciting features in the new release are the replication options. One of the options is a shared disk schema in which multiple IDS servers can share the same disks, dbspaces, and chunks for failover and high availability. In addition to standard High Availability Data Replication (HDR), "Cheetah" offers new Remote Server Replication, which is like HDR over a low bandwidth network. And the previously unsupported version of replication in which you would take the logs from a primary server and continuously roll them forward on a backup server is now fully supported.

In one of the exercises at the training in Toronto, I had a primary IDS server running, two secondary IDS servers sharing the same disks, dbspaces, and chunks as the primary, HDR replication to a fourth IDS server, remote replication to a fifth IDS server, and I was rolling the logs forward on a sixth IDS server. I could fail the primary server, failover, and make one of the shared disk servers the primary, and all the other replication kept going (see Figure 1). I tried several options of failing servers and switching over to other primary servers, and it just kept working. In real life, not many shops would have five IDS servers as backups for a primary server, but this was an exciting configuration to work with. All the servers were active and could be used as read-only servers, so I could make changes on the primary server and see the changes I was making on the backup servers. On the shared disk servers, the changes didn't even have to go anywhere, because the three IDS servers were sharing the same physical disks. Now, it may take expensive disk SAN hardware to implement this, as the disks have to be available and identical on three different physical machines, but that technology is readily available.

Figure 1: IDS 11 Cheetah high availability data replication options.



MORE ON IDS "CHEETAH"

I'll cover many more "Cheetah"/IDS 11 features in future columns. New changes to the sysmaster database and a new sysadmin database will require updates to my sysmaster scripts (which I covered in previous columns) to fully exploit — watch for those in this column. IDS 11 makes this an exciting time for Informix.

Lester Knutsen [lester@advancedatools.com] is president of Advanced DataTools Corp., an IBM Informix consulting and training partner specializing in data warehouse development, database design, performance tuning, and Informix training and support. He is president of the Washington D.C. Area Informix User Group, a founding member of the International Informix Users Group, and an IBM Gold Consultant.

[Return to Article](#)