TeamQuest® Enterprise Database

What It Can Do
&
Things to Consider Before Setting One Up

White Paper
TQ-WP25 Rev. A

Summary

Some performance tools insist that you should centralize your performance database at the system where the performance console is installed. Other tools don't use a database, providing facilities for performance monitoring but not for historical analysis or capacity planning. Still others provide separate databases for each system analyzed. With TeamQuest® Performance Framework you have the best of all worlds. You can use one or more "enterprise databases" to distribute your performance data in whatever way makes the most sense for your environment.

In this paper we will introduce the concept of an enterprise database, briefly explain the advantages of using enterprise databases, and discuss the issues that should be considered when setting up an enterprise database.

This is a short introductory paper. For additional detail on how to use an enterprise database, see "Getting the Most Out of the Enterprise Database." [1]

What You Can Do

If you invest in a license for TeamQuest View, TeamQuest On The Web, or TeamQuest Alert, included in your package is a license for TeamQuest Performance Framework. As you most likely already know, TeamQuest Performance Framework manages a database of performance parameters. With a vanilla install, these parameters are gathered from and stored on the system where the framework was installed. This allows collection of detailed performance information without a significant impact on the network.

However, having a separate database on each measured system does not make it easy to analyze performance across multiple servers. Sure, you can still look at the performance of multiple servers using various TeamQuest products, but cross-system analysis is not as easy with separate local databases as it is when you keep some level of detail about multiple servers in a centralized "enterprise database."

With an enterprise database, you can:

- Include parameters from multiple systems in one performance chart,
- Perform root cause analysis across multiple servers using tools such as the TeamQuest View correlation feature,
- Analyze the performance of an application implemented across multiple server tiers,
- Easily centralize performance data for modeling (capacity planning),
- Archive performance data for long-term analysis,
- Facilitate installation-specific security considerations (for example, with an enterprise database, performance analysis does not require access to each system being analyzed),
- And more.

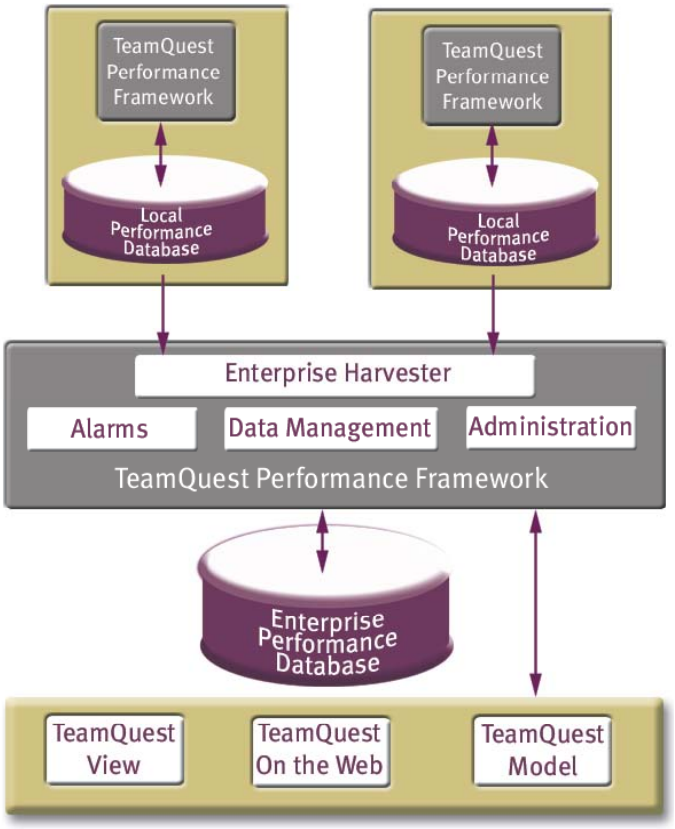

Figure 1
Enterprise Database

Trade-offs to Consider

TeamQuest Performance Framework lets you centralize as much or as little as you need, depending on your intended use of the database. However, the more you centralize:

- The larger your enterprise database,
- The longer your harvests will take,
- The greater the network traffic,
- The more I/O and CPU demands you will make on the system running the enterprise database, and
- The greater the resource demands you will make on the monitored systems as well.

These considerations are not necessarily major limiting factors, it is just that the decision regarding how much to centralize is one of balance and your priorities.

With TeamQuest Performance Framework it is up to you to decide how distributed you want to make your database, depending on requirements unique to your installation. We have some customers who centralize everything they analyze, others who centralize nothing, and a great many who centralize summary information, but leave the details on outlying systems.

Multiple Enterprise Databases

You can have as many enterprise databases as you need. It makes sense to group systems that are being used for similar functions in one database. For example, you might want to keep performance information regarding servers running one application in one database and information about your servers running another application in another database. It also makes sense to keep information for each type of system in a separate database, because sometimes performance parameters with the same name have different meanings on different system types. You might, for example, keep your pSeries servers in one database and your Solaris servers in another.

It is also possible to create a hierarchy of enterprise databases, with higher-level enterprise databases summarizing more detailed data kept in lower-level databases. This solution can be quite useful for larger installations.

You are not limited to a hierarchy of databases organized on a server-by-server basis. It is possible, for example, to keep selected information from the same server in multiple enterprise databases. You can organize information according to departments or business function if you like. It is up to you to choose an organization that makes the most sense for your organization.

More About Trade-offs

Although it is not required, we recommend that you keep enterprise databases on dedicated systems, and/or that you schedule your harvests to occur during off-peak periods. That is, of course, assuming that an off-peak harvest is consistent with your overall goals. (If you want to be able to monitor performance using a single central database in real-time, then you will have to harvest more frequently, though you will probably want to be strategic about which parameters you centralize.)

The key variables affecting the space needed and the CPU resources required for your enterprise database and the amount of network traffic you will generate are:

- How often will information be harvested from local databases to the enterprise database?
- At what granularity will the information be harvested? (Do you want 10-minute granularity, 1-hour, or 8-hour granularity?)
- Which (how many) parameters will you harvest to your enterprise database?
- How long will performance data be retained? (This is only a space consideration, not a network traffic consideration.)

In general, we recommend that you keep detailed performance data aggregated at a fine granularity for a short retention period, and aggregate data more coarsely for longer-term retention. This coarsely aggregated data is a prime candidate for centralizing in an enterprise database, because it can be gathered less frequently, will require less storage space and network bandwidth than data stored at a finer granularity.

An exception to this rule is that you may want to keep Key Performance Indicators (KPIs) at a finer granularity.[4] If you keep KPIs long-term at a fine granularity, then you won't lose information regarding performance dips and spikes that can be lost if you average data points together using a coarse aggregation, and if you choose just a few KPIs, then it can be practical to centralize that information to enable more sophisticated multi-system performance analysis using an enterprise database.

What to Keep

- Details at fine granularity with short-term retention
- Aggregated information at a coarse granularity centralized for long-term retention
- Key Performance Indicators at *fine granularity* kept centrally for long-term retention
- Reduced process information (or no process information) kept centrally

Process information can be very useful for performance analysis, and is necessary in order to use TeamQuest's powerful workload analysis capability, but centralizing detailed process information can generate large volumes of data. We recommend that if you centralize process information in an enterprise database, you should use process reductions to keep the detail at a manageable level. See the "Customizing Workloads and Reductions" chapter of your *TeamQuest Performance Framework User Guide*[2,3] for details on how to set up reduction sets for this purpose.

Conclusion

We hope that this article has helped to identify the issues you need to consider when setting up enterprise databases. We also hope that if you haven't already tried using TeamQuest's enterprise database capability, then we have convinced you that you should.

If you are a TeamQuest customer, then you probably have a license for TeamQuest Performance Framework. With that you get the ability to distribute performance data in a way that is customized for your organization's unique environment. It's powerful, so why not take advantage of it?

Bibliography

1. "Getting the Most Out of the Enterprise Database," TeamQuest Corporation, TQ-WP12 (Dec. 2002).
2. *TeamQuest Performance Framework, UNIX Systems, User Guide,* TeamQuest Corporation, 2001.
3. *TeamQuest Performance Framework, Windows Systems, User Guide,* TeamQuest Corporation, 2001.
4. "Keeping Key Performance Indicators (KPI) for Long Term," *TeamQuest Performance Expert News,* TeamQuest Corporation, Vol. 1, No. 2, (July 2003).

TeamQuest Corporation

Americas
One TeamQuest Way
Clear Lake, Iowa 50428
USA
+1 641 357-2700
+1 800 551-8326
info@teamquest.com

Europe, Middle East and Africa
Box 1125
405 23 Göteborg
Sweden
+46 (0)31 80 95 00

United Kingdom
38 The Old Woodyard
Hagley Hall
Hagley
Worcestershire DY9 9LQ
+44 (0)1562 881889
emea@teamquest.com

Asia Pacific
Level 6, 170 Queen Street
Melbourne, VIC 3000
Australia
+61 3 9641 2288
asiapacific@teamquest.com