

## Software Strategies for On-Site Calibration

Nicholas Mason  
Fluke Corporation  
Everett, WA

### **Abstract**

Many laboratories are using personal computer (PC) based calibration and management stations connected to a centralized database to collect and report information. This system is ideal for fixed computers because they must be constantly connected to a local area network (LAN) to access the centralized database.

But what about mobile computers and hardware that are on the factory floor, or performing calibration on-site at a remote customer? How are changes made on the mobile computer entered on the centralized database? What about changes originating on the centralized system? How are they made on the mobile systems?

Moving data from the mobile system to the centralized system usually consists of crude ASCII importation and loss of detailed calibration results. In addition, information originating in the centralized system often has to be re-entered in the mobile systems because no automated process exists to transfer data back to the mobile systems.

This paper describes a process that allows sharing data among physically distinct databases. This process, called data replication, allows the mobile systems to keep up to date with the centralized system by replicating changes between the databases. Examples shown use commercial calibration and asset management software.

### **Introduction**

In the 60's, calibration laboratories started using automation, but it was in the form of tape drives in the calibrator, or large racks of computer equipment, such as the Fluke Terminal 10 systems. In the 70's, as computers became more common place, labs started using custom computers (controllers), such as the Fluke 1722 or HP 9845.

In the early 80's, the IBM PC became a common computing platform, and was used to setup stand-alone calibration workstations using software like MET/CAL. Over the last decade, computers in a lab or building have been connected using a local area network allowing the computers to share files.

Today, we have a central database on a data server, with calibration workstations networked to the server. All of the calibration and asset data for the standards and UUT's for a given site are stored and accessed from this central database.

But what about different sites that may calibrate the same UUT? A primary and secondary laboratory for the same company may each have their own central database and networked workstations. But if a new device is calibrated by the primary lab, and is recalled by the secondary, the secondary lab has no knowledge that the device was previously calibrated.

We may also have calibration systems that go on-site for an extended period of time to calibrate and collect data about customers' UUT's. This is the same type of problem as the first; we have more than one database that we want to contain the same information.

Problems with multiple databases include:

1. UUT's or standards (assets) may be acquired and initially calibrated (data insert), at the primary site. The secondary site doesn't automatically get the information, and re-enters the data. It also has no history about the UUT that may be necessary to repair or calibrate the device
2. The opposite of 1, an asset is acquired and initially calibrated at a secondary site, but is recalled at the primary site.
3. A change to existing data is made at the primary site (data updated). For example, the wrong serial number or other information is in the database. So the data is corrected at the primary site. If the asset is recalled and calibrated at a secondary site, the wrong serial number will be printed on the certificate, because that database was never updated.
4. The opposite of 2. A change made at a secondary database doesn't get to the primary database.
5. We delete some data at the primary database (data delete). The secondary system will still have the incorrect information.
6. Opposite of 4, data is deleted at a secondary site.

Problems one and two are difficult, but not impossible. Additions to a database can be extracted and inserted into another system. However, problems three through six are very difficult to solve.

Due to advances in technology, there is now a system that solves the above problems by allowing sharing of data among physically distinct database. This process is called data replication.

### **Data Replication**

Data replication is the sharing of data among physically distinct databases. Changes made to data at any one database are, over time, *replicated* to the other databases in the replication setup.

The product MET/REMOTE provides data replication capability between a consolidated database and remote databases. The consolidated and remote databases are MET/BASE databases that may be at the same site as the consolidated database or at different sites. This means the remote sites could be a sister calibration facility, or on-site calibration.

The consolidated database contains all the data to be replicated. Remote databases also contain all the data. Replication is two-way: changes made at the consolidated database are replicated to remote databases, while changes made at remote databases are replicated to the consolidated database, and then to other remote databases, as shown in Figure 1. All changes are replicated to each site over time in a consistent manner, but because of the update time lag, different sites may not be completely up to date at any instant.

MET/REMOTE does not support peer-to-peer replication nor does it combine divergent databases. Remote databases must start with an exact copy of the consolidated databases. Then the replication process can be relied on to maintain consistency between the consolidated and remote sites.

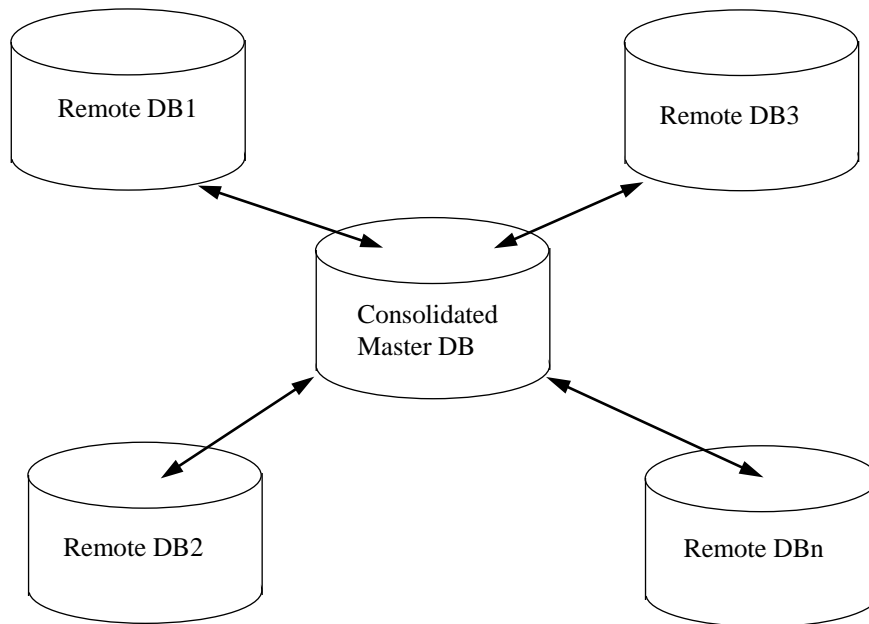


Figure 1: Consolidated and Remote Database Configuration.

### **Publications and subscriptions**

A publication is a database object describing data to be replicated. Remote users of the database who wish to receive a publication do so by subscribing to a publication. Periodically, the changes made to each publication in a database are replicated to all subscribers to that publication. These replications are called publication updates.

Remote databases subscribe to the consolidated database so that they can receive updated information from the consolidated database. MET/Remote always involves messages being sent two ways. The consolidated database sends messages containing publication updates to remote databases, but remote databases also send messages to the consolidated database.

When remote database users modify their own copies of the data, their changes are replicated to the consolidated database. When the replication is made successfully at the consolidated database the changes become part of the consolidated database's publication, and are included in the next round of updates to all remote sites (except the one it came from). In this way, replication from remote site to remote site takes place via the consolidated database.

When the replication is initially set up, the two databases must be brought to a state where they both have the same set of information. This process of setting up a remote

database to be consistent with the consolidated database is called synchronization. Synchronization is carried out manually and essentially consists of making a copy of the consolidated database and copying it to the remote database. Synchronization is used under the following circumstances:

- When a remote database is created. The remote database must start off with a database containing the same information as the consolidated database.
- If a remote database gets corrupt or gets out of step with the consolidated database, synchronization forces the remote site database back in step with the consolidated site.

### **Remote Database Users**

A replication setup includes many copies of the information in a database. While each replicate is a physically separate database on a separate computer, whether it is a laptop computer or a multi-user server, all must stay consistent with the consolidated database. The entire replication setup may be considered a single database, with the master copy of all data being kept at the consolidated database.

Each remote site that submits replications to the consolidated database is considered to be a remote user of the consolidated database. In the case that a remote site is a multi-user server, the entire site is considered to be a single remote user of the consolidated database.

### **Message-based Replication**

MET/REMOTE exchanges data between databases using messages. In message-based communications, each message carries its destination address and other control information, so that no direct connection is needed between applications exchanging information.

Each data server involved in a setup must have a message system agent installed, and they exchange information through the message system. The message system used by MET/REMOTE is a simple file-sharing protocol.

The message replication technology is contained partly in the SQL Anywhere database engine and partly in the Message Agent. The Message Agent is an application that sends and receives messages from database to database. The Message Agent is a program called DBREMOTE, and it must be installed at both the consolidated and at the remote sites.

### **Transaction log-based replication**

MET/REMOTE replication is based on the transaction log, enabling it to replicate only changes to data, rather than all data, in each update.

Recall that the transaction log is the repository of all changes made to a database. MET/REMOTE replicates changes made to databases as recorded in the transaction log. Periodically, all committed transactions in the consolidated database transaction log are sent to remote databases. At remote sites, all committed transactions in the transaction log are periodically submitted to the consolidated database.

By replicating only committed transactions, SQL Remote ensures proper transaction atomicity throughout the replication setup and maintains a consistency among the databases involved in the replication, albeit with some time lag while the data is replicated.

### **Synchronizing databases**

MET/REMOTE replication is carried out using the information in the transaction log, but there are two circumstances where MET/REMOTE copies the entire contents of the consolidated database to the remote database, re-creating the remote database. This process is called *synchronization*. Any data in the remote database that has not been replicated to the consolidated database is lost when synchronized.

Synchronization is used under the following circumstances:

- When a remote database is created synchronization is carried out, so that the remote database starts off with a database in the same state as the consolidated database.
- If a remote database gets corrupt or gets out of step with the consolidated database, synchronization forces the remote site database back in step with the consolidated site.

### **Using MET/REMOTE**

Using MET/REMOTE is not difficult - however, it can be tedious. Many steps are involved and they must be performed correctly or the replication will not work. Fluke provides a preset method to replication.

There are four basic steps to using MET/REMOTE:

1. Install and setup the consolidated database. This step only has to be performed once.
2. Install and synchronize a remote database. This step is only performed when you need a new remote database, or an existing remote database becomes damaged or has not been used for a long period of time.
3. Periodically replicate the database(s). This step should be performed often at both the consolidated database, and at each remote database.
4. Periodically backup the database(s). This step should be performed often at both the consolidated database, and at each remote database.

### **Conclusions**

Many calibration laboratories have a central database where calibration and asset information is stored. One of today's problems is how labs or on-site calibration can share their data with each other. The data to be shared includes new data, changed data and deleted data.

Using data replication can solve these data sharing problems. Data replication allows the sharing of data among physically distinct databases. Changes made to data at any one database are, over time, replicated to the other databases in the replication setup.

## References

<sup>1</sup> Mason, Nicholas, “Providing Data Integrity and Reliability in Your Calibration Data Management System”, Measurement Science Conference: 1997 Workshop and Symposium, MSC, Pasadena, CA, 1997.

<sup>2</sup> Adaptive Server Anywhere Reference Manual, Sybase Inc, June 1998.