Embedded Databases For Embedded Systems

by
Eur Ing Chris Hills BSc (Hons),
C. Eng., MIET, MBCS, FRGS, FRSA

As Published in
Electronic Specifier
February 2015

PhaedruS SystemS

The Art in Embedded Systems comes through Engineering discipline.

Embedded Databases For Embedded Systems

Embedded systems now increasingly need a database. Phaedrus Systems CTO, Chris Hills provides an introduction with links to further reading.

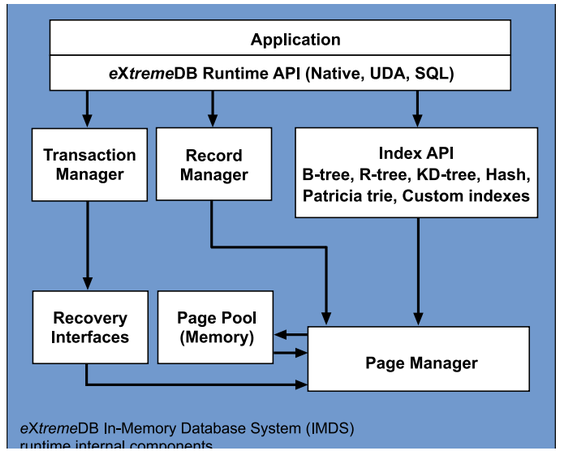
The increase in the power, complexity and prevalence of embedded systems means they are handling more data and consequently increasingly using databases. The traditional solution has been to run this on a separate PC or mainframe in the loop. Today the database can run within the system.

While there are still embedded systems that need only a look-up table or a linked list, for many applications the data demands are such that only a database will work. So how do you solve the problem? Many people think they can write their own database system or use a database not designed from the ground up for embedded systems. Neither route is a good solution.

Writing your own, whether for an RTOS, for graphics, communications stacks or databases, is never a good idea unless you really are an expert. Learning all the techniques and methods takes a lot of time and effort that could be better spent elsewhere on your project.

There is a good summary in this blog of the real costs involved with using "free" software and writing your own at: But it's Free! by Steve Graves Click Link

Developers often think they need to write their own as their problem is unique. When examined, it usually turns out that their problem is not unusual and resolving it is well-understood. Frequently McObject, the supplier of the eXtremeDB in memory database frequently finds that, when examining the database schema and code of a new user that, there is a much more elegant, faster and less complex solution. It's not difficult to come up with a complex solution: the skill is designing a simple and elegant solution that handles the what-ifs. These two



*eXtreme*DB In-Memory Database System (IMDS) runtime internal components

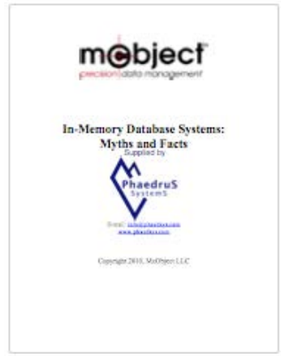
examples alone demonstrate that rolling your own is unlikely to be cost effective, once you have factored in the time to design and build the database and then to debug it.

Once you have decided to buy in a database, you will find that not all databases are the same. Many are designed to run in an environment with disk drives, virtual memory and lots of ever expanding resources (let's call it a computer). There are databases designed from the ground up to work in embedded systems. The most successful route is to use an In Memory DBS (IMDBS). This is not the same as a cached system and IMDBS requires a completely different way of working internally to a conventional DBS. There are many pseudo-IMDBS out there, so it is well worth reading these two papers:

Real vs imitation IMBDS

http://www.phaedsys.com/principals/mcobject/mcobjectdata/Real-vs-imitation-imds.pdf

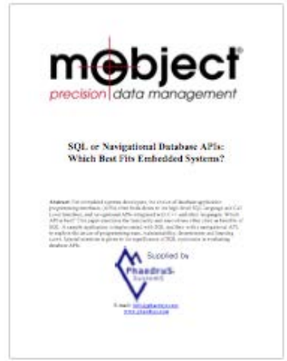
and: IMDBS facts and myths



http://www.phaedsys.com/principals/mcobject/mcobjectdata/imds-myths-and-facts.pdf

You should not start designing your database until you fully understand how an IMDBS works. For example you need to have a strategy for persistent and non-persistent data (And need to know what they are!)

Another choice is of course how to talk to the database. SQL, often the first thought, is a 1970s system that brings it own baggage. For a better understanding of API issues, see



http://www.phaedsys.com/principals/mcobject/mcobjectdata/SQL-vs-Navigational-APIs.pdf .

Like much in system development, adding a database is not easy, but with careful choice of database the overall system will be more powerful, secure and stable.

Embedded Databases

For Embedded Systems

Originally published in Electronic Specifier February 2015

First edition February 2015

Phaedrus Systems Library

The Phaedrus SystemsLibrary is a collection of useful technical documents on development. This includes project management, integrating tools like PC-lint to IDE's, the use of debuggers, coding tricks and tips. The Library also includes the QuEST series.

Copies of this paper (and subsequent versions) with the associated files, will be available with other members of the Library, at:

http://library.phaedsys.com

PhaedruS
SystemS

The Art in Embedded Systems comes through Engineering discipline.