

# FACT SHEET



## Adding value for Software Engineers

### What's the problem?

Software development is a product of the engineer's talent, experience and his knowledge base. There is no substitute for experience, enabled by a natural talent, when it comes to developing high quality code, but if an engineer is tasked with developing software for a protocol or device with which he has no experience, even the best engineer faces a learning curve.

Because of this, the amount of time and effort required to develop embedded software can often be miss-interpreted. The engineer first needs to traverse that learning curve. This is a difficult phase to plan for and underestimating the effort required can lead to delayed implementation and missed deadlines.

### Where is the value?

If the software adds value to an end product, by introducing innovative or advanced functions or implementing cutting-edge developments – and gaining this knowledge strengthens the software team's overall capabilities – this may be considered worthwhile, and the penalty of missed deadlines may be borne in some cases.

More problematic is when the software written adds little or no value to the end product, where only mainstream functions which are commonplace are added, but huge effort for even the most talented software engineering team is still required.

One example could be Ethernet, which is rapidly becoming commonplace in all sorts of embedded devices, even home DVD players. Soon it will not be seen as a value add, but rather a stan-

dard feature which is expected to be included in much the same way as S-Video or Dolby digital decoding have become today.

Initially, adding Ethernet connectivity to an embedded solution may seem like a relatively simple task; specifications and examples can be found on the Internet. Because of this, the Design Cycle may not allow for how long its implementation may actually take. Only once the task is undertaken and the full complexity of Ethernet's protocols and requirements is understood can the effort required to implement it be fully appreciated. By then, schedules have been made and approved, and are expected by management to be adhered to.

Further considerations such as reliability of code, field testing and evaluation all have to be completed, adding to the total cost of developing code which adds no innovation and little value to the product.

Software engineering teams are expensive to maintain and therefore their time is best applied to developing value added features and functions. The reality is, it costs just as much to develop standard enabling software as it does to develop innovative, differentiating code. The difference is, the differentiating code is adding value far in excess of its cost of development, while the enabling code doesn't.

How many projects have gone over budget or have been delivered late because the software team simply had too much code to develop?

Licensing commercially available middleware is now common practice, because it provides functionality that would be too costly or too

risky to develop in-house. The same applies to enabling software; code that for whatever reason isn't available commercially and so must be developed in-house. Proven Software Solutions specialises in sourcing and productising this kind of enabling code so that now, for the first time, there is a viable alternative to developing it in-house.

### **Summary**

By utilising software provided by Proven Software Solutions, time and money can be saved, by integrating pre-evaluated, proven code. Instead of building a working knowledge of complex protocols or devices, the software engineering team need only understand the code's APIs – documented and supplied with the source code.

This allows the engineering team to focus on those parts of the design that differentiate the product in the marketplace, adding value to the end product and saving months of man-effort in the Development Cycle.

---

**Contact Proven Software Solutions today to find out how you could benefit from our services.**

### **Contact details:**

Proven Software Solutions Ltd  
5 Weill Road  
Aylesbury  
Bucks  
HP12 9RH  
U.K.

Tel: +44(0)870 950 3048

Web: [www.proven-software.com](http://www.proven-software.com)

Email: [info@proven-software.com](mailto:info@proven-software.com)