

## Why Embedded GUIs Fail and How to Make Sure Yours Doesn't

### Table of Contents

Executive Summary .....	1
Importance of the Interface .....	1
How GUIs Are Programmed to Fail.....	1
Wind River Tilcon Graphics Suite .....	2
Wind River Tilcon GUI Engine .....	3
Conclusion .....	4

### Executive Summary

Traditional approaches to developing products with embedded graphical user interfaces (GUIs) are time consuming, costly, risky, and engineering driven. Interface development is onerous, involves coding, and is done at the end of the process when changes to the product are impossible or require regression testing of "finished" components. Time, energy, and resources are often spent creating a lackluster interface that isn't eagerly adopted by customers and doesn't drive company sales.

Wind River Tilcon Graphics Suite offers an innovative, more agile approach to GUI design and development. With Tilcon Graphics Suite you can build, test, and perfect an interface without coding and before writing any product code. The portable, proven, and optimized Wind River Tilcon GUI Engine integrates the interface with underlying hardware and software applications. The result is an embedded interface that is developed in a fraction of the time and at far less cost, generating competitive advantage with a superior, more usable interface that drives adoption rates and sales.

### Importance of the Interface

A GUI embedded in a mobile phone, an entertainment system, an appliance, or a scientific or industrial control device can mean different things to different people. To technical professionals who work on the product, coding the hardware and software at its heart, the interface can be an afterthought. It's a minor piece of software and display hardware that is added at the end of the development process to make a sophisticated and feature-rich product usable by the general public. To marketing professionals, the

interface drives sales. It's what gives the product its sizzle. It's the four-color face of their brochures that determines whether consumers buy and the company grows.

The interface determines how easy—and in the case of some control devices, how safe—the product is to use. It is the deciding factor when it comes to how eagerly customers adopt a product.

### How GUIs Are Programmed to Fail

The usability of an embedded GUI can make or break a product. It determines how easy it is to use the product, how well it works, and how enthusiastically it is adopted by consumers. Along with cost of development and time-to-market, it also drives the success of product manufacturers. Apple's iPod and iPhone are good examples of how an attractive and easy to use embedded GUI can generate market share. Nothing is more frustrating than seeing a competitor's product with inferior components and a shorter, less impressive list of features eat into your market share because its interface makes it easy to use, generates positive reviews in trade journals, and excites buyers. Yet traditional methods of developing embedded GUIs almost ensure they will not become killer interfaces that consumers demand and manufacturers dream about. Why?

To understand where the usual development approach falls short, here is a look at the traditional way to develop a product with an embedded GUI.

It starts when the marketing team, based on sales data, market research, or customer requests, sees an opportunity to meet consumer demand. They outline the need, describe the product that can meet it, and pass their marketing specifications along to the company's engineers.

Based on marketing's list of use cases, features, and other requirements, a hardware and software architect, or a chip architect, assembles engineering specifications. Guided by these specs, a highly skilled team of hardware and software specialists sets to work laying out the chip, integrating its components, and ensuring process timing. Members of the team do signal processing, create algorithms, build the required housekeeping routines and low-level software, and perform the many other tasks needed to make real what marketing and engineering have envisioned.

When all of the pieces are assembled into a prototype, integration testing begins. Each process and component in the product is tested thoroughly to ensure that it works as required, both on its own and in chorus with other components. Can the device stay on for eight hours at a time? Does it automatically shut down when it detects a low battery? Does it properly decode 5.1 surround-sound Dolby audio? These are the types of questions asked and answered during integration testing.

With testing complete, and everything working according to the specifications, the product is almost ready to go. All that remains is to add the GUI. Building that GUI and making it part of the product involves its own coding and other tasks. But there are code libraries with widgets, dials, boxes, drop-down lists, and more to help speed the process. And graphic artists, designers, and usability experts can help make the GUI friendly and usable.

When the coding is done and tested, the interface, and the product, is ready to go. If adherence to marketing and engineering specifications is any indication, it is sure to take the market by storm.

Of course, in reality it's not that simple. Between marketing's initial idea and the finished product a lot of time passes, and a lot can change. Subtle differences between what was asked for based on consumer demand and what is delivered because of translation issues and the realities of real-world development can make big differences when it comes time to market the product.

The development process itself may well take longer than planned. And chances are good that the GUI, which to the customer is the product, has consumed a larger chunk of time than expected and its usability and appeal have suffered from its position at the end of a lengthy process.

True, code libraries do make the job of developing a GUI easier. But libraries still have to be linked to the product. And once developers choose a widget, dial, or list box from a library, they still have to generate code to specify its size, direction, color depth, and other factors. They also have to test and validate the code until the interface works properly.

Once you have an interface that marketing, engineering, and possibly quality assurance and test customers can start using, you're still not finished. Inevitably, input from artists, designers, and usability experts, assuming they are available, results in changes to the interface. These professionals are likely to point out that it's too late in the development process for them to make a real difference—that they should have been involved from the start. At best their changes mean repeating the cycle of interface coding, testing, and validation. More likely they also require changes to the product hardware and

underlying software, which means repeating the integration testing that you thought was completed, possibly more than once.

Changing a single line of code in the interface can easily require revalidating the entire system through several layers of complexity, dramatically increasing development cost and time. More important, the chance of creating an embedded GUI that lifts your product above the competition steadily diminishes, as the interface is increasingly crafted to meet engineering specifications instead of to ensure usability and appeal.

Given the elaborate and complex process needed to create the underlying product, it should come as no surprise that an interface built at the end of that process bears all the marks of software developers and engineers who thrive on complexity and who understandably want it reflected in a sophisticated device. The result is more often than not a product that is engineering-centric and not user-centric—a product that doesn't excite people and isn't adopted by them.

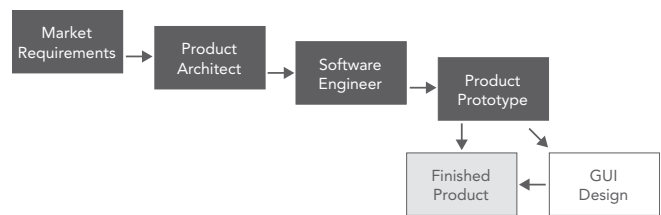


Figure 1: The traditional product development process

### Wind River Tilcon Graphics Suite

Wind River Tilcon Graphics Suite offers an innovative user-centric model for embedded GUI design and development that lets you build high-quality embedded GUIs in a fraction of the time at far less cost. By streamlining and improving the development process and making GUI design and development faster and easier, Tilcon Graphics Suite dramatically shortens time-to-market and improves the end result, enhancing the user experience to drive higher adoption rates and sales.

Unlike traditional approaches, Tilcon Graphics Suite ensures that the design and development of an embedded GUI is not left until the end. Instead it is done early in the process. In fact, the GUI can essentially be ready to go before a single line of product code is written.

What makes this possible is Tilcon Graphics Suite's interface construction tool. It brings GUI designers into the process from the start, engaging them early on so they generate a vision for the interface based on its functional specifications

and can quickly build that interface, share it with stakeholders throughout the organization, and perfect it to reap the benefits of a superior design. In the process they reduce the length and complexity not only of interface development but of the overall product development process. This also means rigorous usability testing and improvement can be done before a single line of code is written. This leverages usability best practices and makes for the kind of drop-dead simple interface that breeds excitement and success.

How does this work? When the development project is in its earliest stages and marketing and engineering are still scoping out the technical details of what will be needed to make the product great, GUI designers and usability specialists can generate graphical interface components in Adobe Photoshop—the tool of choice for much of the creative community—and convert those components to a working prototype with the Tilcon Graphics Suite interface builder.

What this means, of course, is increased emphasis on the usability—and the excitement factor—of the product interface. It provides ample opportunity at the very beginning of the project to ensure that the GUI you embed in your product is the best it can be. Input gathered from stakeholders can be used to constantly improve the usability and appeal of the interface throughout the development process, while keeping control firmly in the hands of design experts until the result is a highly effective interface that not only works well but provides a compelling experience. Before you’ve spent a dime on hardware and software coding, the interface has been socialized throughout the organization and perfected, based on all of the relevant input. Instead of trying to fit a square peg into a round hole at the end of the process, you start with an interface that is an integral part of the product.

Perhaps more exciting, this prototype interface is not just the starting point for GUI development; it is the interface itself. When it is finalized, it is ready to be embedded in the product. All the messy steps associated with getting a GUI translated and iterated, of continually degrading its usability to meet engineering specifications, are gone. This is exciting simply from a GUI perspective. But it also has major implications for the product development process as a whole. Recall that with the traditional method of GUI development, changes at the end of the process can necessitate lengthy and expensive regression testing. With the GUI developed upfront using Tilcon Graphics Suite, your development process is faster, less risky, more efficient, and more cost effective.

## Wind River Tilcon GUI Engine

Instead of buying a code library, attaching it to your product, and using it as the starting point for coding interface components, you can simply attach Wind River Tilcon GUI Engine to your product. The engine is a run-time platform that sits on the product and integrates all product applications with the interface.

The Tilcon GUI Engine displays your GUI on a liquid crystal display (LCD) screen. It communicates with the product, whether it’s an MP3 player, a DVD playback device, an industrial controller for a conveyer belt, or anything else, and runs its applications and processes while displaying the results for the user.

Tilcon GUI Engine can work on a wide range of architectures and develop custom engines and application program interfaces (APIs) as needed. Simply attach the portable, proven, and optimized Tilcon GUI Engine to your product and it integrates the interface that you’ve developed with the underlying hardware and software for a complete, working product.

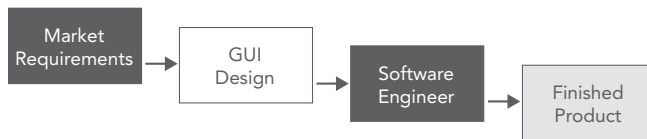
Event	Using Traditional Code Generation Methods	Using Wind River Tilcon GUI Engine
View and Test the Interface	Compile the Graphics Code; Then View, Test, Change, and Recompile	No Coding Needed; Graphic Objects Are Built Visually and Tested in Real-Time
Change the Underlying Hardware or Software	Port and Reoptimize the Graphics Code	No Porting Required; Graphic Objects Run Anywhere Tilcon GUI Engine Runs
Rebrand or Reskin the Interface for a New Customer	Rebuild the Application, Test It, and Download It to the Target	No Application Rebuilding Necessary; Change Any Graphic Element and View It Immediately on the Target
Test and Perfect the Interface	Make Code Changes, Introducing Risk to the Entire Project and Taking Time and Effort to get It Right	No Effect on Underlying Application Code; Graphics Changes Can Be Made Without Risk or Increasing Time-to-Market

Figure 2: Traditional interface code generation vs. Wind River Tilcon GUI Engine

## Conclusion

What does Wind River Tilcon Graphics Suite mean for developers of products with embedded GUIs? By bringing interface design to the fore instead of leaving it till the end, you ensure that the GUI you embed in your product is designed from a usability perspective: It is user-centric, not engineering-centric. It embodies the creativity and vision of talented and knowledgeable designers and artists, while meeting the exacting specifications of marketing and engineering. It is not only complete and effective, but friendly, highly usable, ergonomically sound, and aesthetically appealing. And it's ready to go as soon as you've completed integration testing.

With the Tilcon Graphics Suite approach, embedded GUI development starts early—before you write a line of product code. It is faster, less risky, requires no coding, and ensures highly usable interfaces that generate excitement.



*Figure 3: The Wind River Tilcon Graphics Suite product development process*

Embedded GUI development with Tilcon Graphics Suite simplifies and reduces the risk associated with creating a GUI. It replaces an onerous coding and recoding process with a straightforward and collaborative approach to interface design that streamlines overall product development and ensures a positive user experience.

Wind River Tilcon Graphics Suite makes life easier for everyone involved in development. And it generates the excitement and brand loyalty that comes from a superior, more usable graphical interface.