

## CHARTING COMPONENT STRENGTH - FINDING THE RIGHT CHARTING TOOL

CONGRATULATIONS! Reading this page means you're likely a product manager, product owner, or should be one.

### CLIENT- SIDE

If you are creating a client-side EXE working with: .NET WinForms, MFC/ATL, VB6, Delphi, Builder, NI's Measurement Studio or Labview; stop searching. You have found the correct charting component. Download our eval and hammer our PeDemo for at least 30 minutes. Then study the code in the demo for at least 30 minutes. You won't find a more impressive demo or easier code to comprehend anywhere in the world. The bits don't deceive, and they are our best selling point. Like a picture is worth a thousand words, our PeDemo is like an infinite amount of pictures. Sizing, shaping, using the popup menus (right click) and dialogs allows for infinite variety and massive exposure to our bits. PeDemo's intelligently dynamic behavior is very impressive. Nothing compares; that is why we are the number #1 selling client-side charting component.

When launching our PeDemo, Please read the demo's "About Box". It seems trivial but there are a few key pieces of data, like automatically launching our help system from the code window. Thank you for reading the "About Box".

Occasionally we hear a remark that ProEssentials is not 100% managed code. Thinking like a product manager, you should prefer ProEssentials to "not" be 100% managed. Our charting tool works perfect in .Net applications. It may not have some .Net centric features, but what it lacks is not affecting the problem/solution domain. ProEssentials being native code only increases its potential to solve your problems. Our .Net interfaces are not object crazy like most .Net components. We know our .Net interfaces are easiest to learn and maintain. High object use within an API only leads to higher abstraction. Our interfaces are simply 95% properties, 5% methods, and events. No object craziness; just set properties. Managed code was invented for servers. To extrapolate the concept into a system level DLL is absolutely wrong. We consider ProEssentials a mission critical system level DLL, and is designed as one. Managed code may be fine for some of our competitors producing glorified charting buttons, but a real charting component should be designed like a system DLL. MS is now starting to add more new features to VS targeting native development. You won't see MS or Adobe ever writing system level DLLs in managed code. As a VSIP, Gigasoft gets solicited by MS to create new managed interfaces to newly evolving native system DLLs. Native code is not, nor will ever disappear, and will always represent the common denominator for producing the highest quality software. ProEssentials can selectively use GDI or GDI+. ProEssentials can produce the same look as a 100% C# component, while optionally charting in GDI when speed is necessary. ProEssentials is magnitudes faster than any 100% C# charting tool. For producing client-side EXEs needing charts, nothing is better than ProEssentials.

## SERVER - SIDE

This is a big subject so I will break it down: 1) Traditional browser based apps; 2) Internet aware client side development (Google Earth / ITune type EXEs)

### 1) Traditional Browser Apps:

Due to our a) client-side strength and b) strength of rendering quality/robustness, ProEssentials within the browser diverges towards two paths:

a) The path toward absolute maximum interactivity, speed, robustness, but limits browsers to IE / Windows OS.

b) The path toward minimum interactivity, excellent image quality independent of size, robust graphic potential, and works with any browser.

a) Maximum Interactivity: Our .NET WebForm controls render the binary state of our ActiveXs onto the client's browser (no dependency on COM on the server.) Check out our online demo for a working example. This technology does require IE and Windows OS and thus best suited for specific web solutions to specific customers. For example; Siemens, Square-D, and Schneider Electric use our technology to allow customers to monitor their large scale power meters.

Financial apps are also a great fit for this technology as our online demo demonstrates. Your server logic can always conditionally send browsers either ActiveX or Static Images based upon browser capabilities and/or user's choice. If sending static images, you're still getting maximum graphical potential to the largest problem domain. There are charts in the industry that use AJAX to gain some form of interactivity. Please compare all other interactive demos you can find on the web to our online interactive demo. Comparing AJAX to true interactivity is like comparing real auto racing to an XBox racing game, they're worlds apart.

b) Minimum Interactivity but maximum audience: Our .NET WebForm controls rendering simple JPG or PNG images with or without imagemaps (hotspots.) Again, you'll be getting the maximum graphical potential to the largest problem domain. Features such as multiple axes, date-time, annotations, and overall intelligent image construction help maintain chart quality at smaller chart sizes. If you have a huge site producing thousands of images, nothing uses fewer clock cycles per image than ProEssentials. ProEssentials does more, with more intelligence, with fewer cpu cycles.

### 2) Internet-Aware client side development:

I always recommend to develop internet aware EXEs over browser apps. The browser was originally designed for static words, images, and simple form processing. I prefer Google Earth over Google's web maps. iTunes and Napster work better as an internet-aware EXE than would as a web site. If you're working with lots of data, want maximum interactivity, much easier development, better testing and debugging, better performance, more stability, and so many other advantages, create a simple EXE that gets its data over the net. ProEssentials has very simple features to 1) efficiently transfer binary chart states across the net, and 2) incrementally load data across the net. Our WinForm interfaces, native DLL, and ActiveXs can all receive data from our WebForm interfaces. We provide all interfaces to allow all potential combinations of EXE, Browser App, and Internet-Aware EXE development. Internet-aware EXEs are the best; and ProEssentials with its client-side strength is the best chart for your internet-aware EXE.

THE BOTTOM LINE (It's about pleasing and protecting the customer, and what we do best!)

Our customers (thousands) are largely software-business-owners and high-level product managers whose direct bottom-line are based upon the charting functionality embedded in their product(s). When compared to all potential buyers in the market, our customers have spent the most time evaluating their choices because their needs are the most important. When it comes to our customer's money, reputation, and product competitiveness, ProEssentials is being chosen. One of our greatest strengths is the type of customer we attract.

Gigasoft's business plan is focused on the customer's success by offering more and more quality features rendered with maximum commercial-grade/oem-type attention to detail. This is obviously not the most profitable strategy as our largest selling point is only revealed after you spend time with our tool. The more profitable strategy is to focus on buzz words, writing components with limited content, solving multiple problems in a limited way (component suites,) but a few great selling features, concentrating on designers, wizards, and a fancy web site. Please call me an engineer and not a business-man because I choose to take an engineering approach to business over a pure marketing approach. When producing lighter-weight tools, all these designers and wizards is much more manageable. It's funny that the industry authors calls this type of product "full-featured"; when in fact, the true "functionality embedded" is anything but full-featured. I sincerely hope you see through the light-weightness and seductive curb-appeal of our competition. The faster Microsoft continuously changes the face of Windows development, the more it forces component tool vendors to supply lighter-weight tools. The more lighter-weight tools, the more negative impact upon the component industry. Within this turmoil of constant change, Gigasoft's business strategy counteracts this natural evolution towards producing lighter-weight functionality. For over 12 years, our customers have always been able to upgrade easily and at their own pace. We care more about maintaining and improving our customer's code base than forcing change and chasing fads. Our customer retention is perfect, our customers upgrade even though not forced to upgrade. Word-of-mouth and repeat-customers are our largest source for new sales. We constantly get new customers switching from our competitors. I've never heard of losing an existing customer to a competitor. If you buy ProEssentials; you will be satisfied, you will be a customer for life, your applications will easily and inherently improve in quality over time, our attention-to-detail means you'll work much less making your charts behave as they should, and you will eventually see the competition clearly secondary. This is my personal promise to you as a fellow engineer, in friendship and character.

Thank you for reading this very long letter. I sincerely appreciate your dedication in researching your purchase.

Best regards,

Robert Dede  
Owner/President  
Gigasoft, Inc.  
[robert@gigasoft.com](mailto:robert@gigasoft.com)

Gigasoft is a registered trademark, and ProEssentials a trademark of Gigasoft, Inc. Microsoft, Xbox, SolarWinds, Network Computing, Adobe, Delphi, Builder, Google, Google Earth, iTunes, Napster, Siemens, Square-D, Schneider Electric, Measurement Studio and Labview are trademarks of their respective companies.