THE PREEMPTIVE ANALYTICS DIFFERENCE:
Fifty Facts & Features

Development Controlled Runtime Data Transmission

Avoid slowing applications, burdening networks, clogging data stores and impeding analysis with noise.

1. **Instrumentation:** The art of including technology and logic inside your app to transmit meaningful usage, quality and adoption telemetry safely, securely, efficiently, and reliably.
2. **Injection:** Dotfuscator and DashO can inject instrumentation into your apps AFTER they've been compiled. You can not only instrument apps your writing now but any app you've ever written.
3. **API SDKs:** For greater control (and for runtimes where injection is not feasible like JavaScript and native), we provide developers with the APIs they need.

Application-Specific Data Type & Field Support

Avoiding the "one-size-fits-all" approach that diminishes application analytics value.

4. **Session Data:** You can easily gather session counts, time spent inside an app, adoption rates, geolocation, etc.
5. **Performance Data:** You can easily gather memory usage, CPU consumption and other performance related statistics inside any section of your code.
6. **System Data:** You can detect screen resolution, sound chip manufacturer, available drives, etc.
7. **Feature Data:** You can track feature usage, sequence, time spent inside a feature, etc.
8. **Custom Data:** You can gather any piece of data that your app can see from a serial number to the string typed by your user into a search sliver.
9. **Unhandled Exception Data:** You can gather stack trace and other supporting data.
10. **Thrown Exception Data:** You can track and monitor any thrown (handled exception) like "SQL Injection detected."
11. **Caught Exception Data:** You can track and monitor any caught (handled exception) like media not supported or file not found.
12. **Exception Dialogue:** You can prompt the user for additional information, contact details and opt-in/out permissions.

Seamless Integration into Development Platforms & Operations Environments

Avoid unnecessary quality, service and operational risk and expense.

13. **Team Foundation Server Subscription:** You can send exception data to our rule-driven service that will automatically create Team Foundation Server work items.
14. **Visual Studio Integration:** Want to preserve a seamless integration with Visual Studio? Again, not a problem.
15. **MSBuild:** We offer a first class MSBuild task.
16. **Command Line:** Of course we can be called from a command line.
17. **Stand-alone:** Don't have a seat of Visual Studio to spare? Not a problem.
18. **Distributed Development Support:** Need to do continuous integration or at least distributed across groups? We're built for that.
19. **Delayed Signing Capability:** Need us to sign your app when we're done? It's already built in as an automated feature.
20. **ClickOnce Support:** ClickOnce is complicated to transform but not for us!

21. **Silverlight and WPF Support:** Built-in.
22. **.NET and Java Release Targeting:** Working on one version but you need to target another? Piece of cake – we will not break your app.
23. **100% .NET Compliance:** Need to PE Verify? Looking to deploy on Mono? It's all good because we break nothing!

Automated Privacy & Security Policy Enforcement

Simplifying development and reducing operational and regulatory risk.

24. **Opt-in Policy Enforcement:** Our instrumentation can automate and enforce your opt-in policy.
25. **Opt-in Dialogue:** Further, you can customize the dialogue and/or the logic.
26. **Opt-in for Sessions:** We have one opt-in setting for healthy use (customer experience monitoring).
27. **Opt-in for Exceptions:** We have another opt-in setting for exception processing and monitoring.
28. **Encryption for Network Traffic:** Our instrumentation encrypts your data on the wire for increased security.

Platform, Device & Language Support

Ensuring that wherever your code needs to go or how it's developed, your view into its adoption and impact will remain unimpeded.

29. **.NET, Windows Phone & WinRT:** Any MSIL – anywhere.
30. **Java and Android:** Any JAR file – anywhere.
31. **JavaScript:** Running anywhere from Windows to WinRT to iOS.
32. **Native Win32/64:** For high performing and specialized applications.

Performance & Stability

Ensuring that analytics don't cause more problems than they detect.

33. **Offline Caching:** Our instrumentation automatically writes to isolated storage when there is no network available and sends cached data when it can.
34. **Mobile Optimization:** Our instrumentation is optimized to preserve bandwidth and battery life.
35. **No Agents Required:** Our instrumentation requires no agents, third party libraries or any other artifact – all instrumentation lives inside your applications.

Visual Studio & Team Foundation Server Integration

PreEmptive Analytics was first embedded inside Visual Studio with VS2010. With VS2012, our role has been expanded dramatically to include an on-premises runtime data repository and a "production incident aggregator" that applies rules to collected exceptions and automatically generates work items when those rules are triggered. *Why should this unique and special relationship matter to developers?*

36. **The Most Rigorous Quality Criteria:** PreEmptive Analytics is subjected to Microsoft regressions tests, security audits, code reviews and quality gates – certified by Microsoft – as an organic part of the Visual Studio development and release process.
37. **Integration into ALM and DevOps Patterns and Practices:** Our close collaboration with Microsoft's development organization ensures that our Agile conventions, data structures and taxonomy are not only compatible – but deliberately extend – Visual Studio's commitment to Agile development.

38. **Upward Compatibility:** PreEmptive Analytics will never slow you down. PreEmptive Analytics offers the broadest .NET, Java, JavaScript and native support: due to Microsoft integration process, PreEmptive Analytics is integrated into every version of Windows and the .NET framework – no matter how early your access to their "bits."

39. **Hardened Against Hundreds of Thousands of Real-World Scenarios:** While every development team's requirements are potentially unique an active install-base estimated at over 250,000 – you can be pretty sure that every defect has been identified, diagnosed, and addressed.

40. **De Facto Development Standard:** A large active install-base bolstered by published patterns and practices and tight integration with both Visual Studio and Team Foundation Server means that your investments in Dotfuscator protection and instrumentation will be preserved and ultimately amplified by your core investments in Microsoft development tools and platforms.

Security

Deterring the opportunistic, impeding the malicious and enabling the prosecution of the successful is what application security is all about. Through a combination of preventative controls and detective controls, PreEmptive Solutions has helped to secure and extend application intellectual property integrity and value.

41. **Patented Renaming Algorithm:** Induction overloading takes renaming to a new level of obfuscation. No other supplier can offer this technique.

42. **Control Flow:** Our experience with optimization makes our control flow effective against machine translation while minimizing performance impact.

43. **String Encryption:** Our work with various defense departments makes our approach secure and legal for export.

44. **Built-in "Smart" Obfuscation:** Provides optimized support for XAML, BAML, XAP, mobile and ClickOnce scenarios to name just a few.

45. **Tamper Detection:** Obfuscation is designed to prevent reverse engineering and tampering. Our tamper detection and defense is a "detective" rather than "preventative" control that is virtually impossible to detect let alone remove from your application.

46. **Tamper Defense:** This service can mount custom, real-time defense mechanisms that are 100% customizable by your development team.

Customer Support

Post compile obfuscation and the injection of instrumentation into your precious code is no small task and typically happens at the very end of your production lifecycle – where time and a sense of humor are often in equally short supply. That is why we focus on managing your risk and accelerating your work with a hands on approach.

47. **Dedicated Customer Support Staff:** Our dedicated, professional team knows how to program, they know our technology and their entire job is to make you successful.

48. **Live Support:** Our dedicated team will actually chat or speak with you live on the phone.

49. **Enterprise-Ready SLA Capacity:** We have the ability to commit to organization-specific SLA's allowing you to transfer risk from your organization.

50. **Online and Newsgroup Support:** We post our (and our clients') tribal knowledge online so you don't have to reinvent the wheel.

The PreEmptive Difference

Over the past 15 years, our team has focused on protecting, monitoring and measuring application value. With over 4,000 clients and software on millions of developer desktops, there can be no question that organizations that are serious about building and securing the best software possible trust PreEmptive Solutions. Here are a few final reasons why.

51. **Category Creators:** We don't just lead in the obfuscation, application risk management and application analytics markets – we literally created them.

52. **Proven Quality and Service:** The largest manufacturers, life science companies, aerospace corporations, financial institutions, and (of course) software development organizations all trust PreEmptive Solutions to secure and measure their work without compromising the quality and functionality of their code.

53. **Cross Platform:** From Azure to Silverlight to WP7 to the next great thing – we support these frameworks, platforms and surfaces on the day they arrive.

54. **Exceed Expectations:** Just as we delivered 54 reasons to work with PreEmptive Analytics when we only committed to 50 – we will exceed your expectations for quality, responsiveness and capability.

1999
DashO Launched

PreEmptive Solutions establishes obfuscation category built upon optimization expertise.

One of our largest Java clients at the time: Microsoft Corporation

2006
Detective Controls
Tamper Defense and Shelf Life

Establised app risk category integrating technology, process and controls.

2010
Instrumentation Included in VS2010

PreEmptive injection of instrumentation, tamper, and shelf life established as de facto standard for .NET framework

2003
Dotfuscator Launched

Dotfuscator Community Edition established as the de facto standard inside Visual Studio 2003.

2008
Runtime Intelligence Service Launched

Established **application analytics** category built upon optimization and post build expertise

2012
PreEmptive Analytics for TFS Included with VS2012

DevOps solution simultaneously released commercially for VS2010 and inside VS2012, connecting production incidents to development and operational workflows