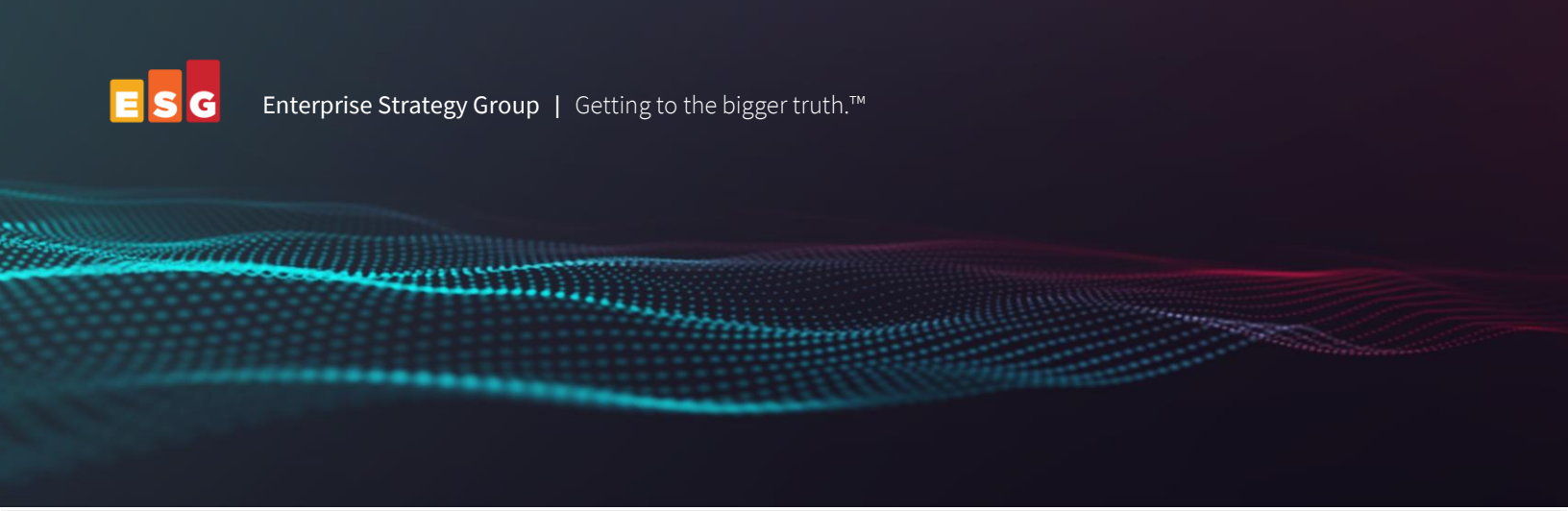

ESG WHITE PAPER

Oracle Database 21c and APEX Service Deliver its 'Magnum Opus'

An Overview of Oracle Database 21c's New Features and Low-code Development Using Oracle APEX Service with the Oracle Autonomous Database

By Mark Peters, Principal Analyst & Practice Director; and Mike Leone, Senior Analyst

January 2022

Contents

Executive Summary

Organizations don't just want access to more data, but to more interesting data. And that can be a challenge—is delivering that an art, science, or a symphonic composition of both? As executives and end-users, we expect interesting data to be timely, relevant, and highly contextual, and that takes ever newer and more engaging applications. Our developers choose from a diversity of data technologies, each specializing in its own "harmonies"—databases, messaging, object stores. In the end, though, it is not just a data problem, it's a "right data at the right time in the right format" problem. And composition is just hard; what's needed is a talented conductor aligning all of instruments of the orchestra.

There has to be a better orchestrated approach to addressing everyone's data wants and needs. Of course, while databases are at the center of business transactions, one might be forgiven for assuming that they are difficult to deploy, integrate, and use. That, however, is not the universal state of affairs but it is precisely the problem that Oracle is solving.

Oracle's converged database strategy simplifies, improves, and enables the data-obsessed future that organizations not only desire, but also strive to deliver. And when paired with a cloud-based low-code development platform (its Application Express (APEX, for short) service), users can enjoy rapid application development. This pairing can be viewed as Oracle's "database magnum opus," a symphony of data instruments conducted as one across many and varied organizational "themes" and "movements."[1]

- Oracle's overarching database strategy remains that of convergence. Andy Mendelsohn (Oracle EVP of Database Server Technologies) often uses the deceptively simple—but also effective—analogy of smartphones. Whereas we used to have multiple devices (that is, phones, cameras, pagers, MP3 players, GPS systems, and so on), smartphones converged them into one device for ease, economy, and productivity. The notion of a converged database is exactly the same. Oracle is converging many different database technologies that are frequently offered by other vendors as separate databases into one single database engine that Oracle calls a converged database.

- Oracle Database 21c is the latest version (with 200 "innovations") of Oracle's eponymous database. It further enhances the ability for different data models, data types, and workloads to all play nicely together…and, in so doing, it can increase Oracle Database users' ease, economy, and productivity. There are myriad improvements; some (such as tamper-proof blockchain tables and the use of persistent memory, which are explained in greater detail in the main body of this paper) being so valuable that many of them have been "back-ported" into 19c, which is Oracle's most recent long-term database release. Oracle Database 21c is available on the Oracle Cloud Database Service, the Autonomous Database Free Tier Service, and on Linux platforms including Oracle Exadata.

- APEX is a low-code development tool that has been around (aka refined, proven, and ruggedized) for more than 15 years, garnering, according to Oracle, over 500,000 developer users along the way. APEX lets developers declaratively build applications without laborious coding. Mendelsohn describes that its speed and ease of use are why it's used to build 6,000 new applications every day. The power of Oracle APEX is available as a browser-based, standalone service in Oracle Cloud Infrastructure (OCI).

[1] The title of this ESG white paper, together with elements of the Executive Summary and Bigger Truth sections herein, are adapted from an ESG Blog covering the Oracle announcements and originally published on 01/14/21 (*Oracle's APEX Enhancements & Database 21c Release Continue Progress on its Magnum Opus*).

Market Overview and the Challenges of Data(base) Silos

The top data-centric technology leveraged across organizations' data pipelines today remains the same as it has been for many years prior—the SQL database. In fact, 37% of organizations report leveraging SQL databases as part of their analytics initiatives.[2] But over the years, driven both by IT possibilities and business demands, organizations have looked for ways to incorporate more data *and* different types of data that require different data models. What began as an analysis of just well-structured, relational data has evolved to incorporate analysis of semi-structured and unstructured data. There's text, and XML, and JSON, and spatial data; and then there's relationship-centric data best suited for graphing. You have likely heard of the "data silo" problem. It's a challenge exacerbated by organizations turning to single-purpose data stores (which often end up effectively as isolated *database silos*) for each use case that comes along. And that leads directly to a data fragmentation problem that is ripe for correction.

The fragmented approach of single purpose datastores has, simply, created yet another silo problem associated with operational management. ESG research gives us an idea of the scale of the database silo challenge: 74% of organizations said that in order to yield reliable outcomes in support of data-focused initiatives they had to integrate a minimum of *four* disparate data sources, and that could very well be four different databases; to make this worse, 39% of organizations said they required a minimum of *seven* integrated data sources.[3] And the fragmentation could have a profound impact on the rest of the business, including such issues as constant data movement and data transformation, inconsistent data, security holes, slower application development, resorting to analyzing—and taking action based upon—stale data; the list could go on. The IT, organizational, and business complexities, cost, and risk should not be taken lightly.

The data integration and operational silo conundrum that organizations face today can easily be extended to workloads and access too. Some independent workloads interact with data in different ways, whether those workloads are transactions (OLTP) and analytics queries (OLAP) or modern workloads rooted in streaming data, machine learning, IoT, and even blockchain. There is also an increase in the number of developers who are looking to integrate microservices that require access to real-time events through APIs and REST interfaces. These developers are looking for improvements in building and maintaining reliable CI/CD pipelines. And the latest trend of low-code/no-code is enabling a whole new generation of developers and LOB analysts to add value. But do organizations really want to build out numerous unique approaches that are in turn dependent upon a slew of underlying data stores and their corresponding infrastructures, not to mention hire staff with the data management skillsets needed to leverage those approaches? Oracle's bet is that they don't, and with that in mind, its converged database approach is simply IT common sense leading to uncommon business value. And it comes with a fully integrated, low-code development platform for building, deploying, and maintaining data-driven applications.

[2] Source: ESG Survey Results, *The State of Data Analytics*, August 2019.
[3] Ibid.

Oracle Converged Database 21c: Removing Data(base) Silos

Oracle's converged database strategy is intended to give organizations—whether expert data analysts, novice developers, or anything in-between—a *single view of data through one converged database*. The database can support multiple data types and workloads and provide consistency in how that data is accessed and leveraged for modern applications via open standard SQL. For organizations embracing the cloud, Oracle continues to emphasize the importance of automation through its innovative approach with Oracle Autonomous Database, a 100% self-driving database that leverages machine learning and internal best practices to eliminate the complexities of operating, maintaining, securing, tuning, and scaling a database. But Oracle is not resting on its laurels. It recognizes that to better enable organizations to use more data more effectively, inevitably more and better integrations with databases are essential. The APEX service shines a spotlight on the need for, and Oracle's ability to enable, organizations to build and iterate on modern, data-centric applications more rapidly. By integrating an enterprise-leading low-code environment with the maturity of the Autonomous Database, organizations are set on a transcendent path for faster application development anchored in rich data experiences.

Oracle Database 21c is the latest "innovation release" of Oracle's converged database and offers hundreds of innovations to further support its multi-model, multi-workload, and multi-tenant value propositions.

Some selected highlights included in 21c are:[4]

In-database JavaScript

The de-facto standard scripting language used by myriad organizations in web and mobile application development is now able to execute where the data resides: in the database. Developers can now create code snippets that can execute tasks without requiring any movement of data between tiers or within the browser cache. It runs on the embedded Graal Multilingual Engine (MLE) and enables automatic data type mapping between JavaScript data types and Oracle Database data types. By leveraging the plethora of open-source JavaScript libraries and easily executing SQL from JavaScript code, developers are enabled to work efficiently in a scripting language that matters to them.

Native JSON Data Type

While JSON has been a part of Oracle Database for years, 21c offers organizations JSON as a native data type. In other words, there are no more performance penalties for read or update operations that previously required JSON parsing; instead, only inserts require parsing, but once inserted, the new internal binary format enables faster scan and update performance. In fact, Oracle expects scans to improve by up to 10x, and updates up to 4x.

Blockchain Tables

Oracle understands that one of the greatest barriers to blockchain adoption is building applications to effectively utilize a distributed ledger. In 21c, Oracle has deeply integrated blockchain into the Oracle Database, making it possible to implement blockchain in mainstream applications with minimal changes to the underlying application. The core focus of Oracle blockchain is to prevent illicit changes to data that contain key actions, assets, entities, and documents. Oracle offers crypto-security technologies that offer organizations the ability to incrementally adopt them based on a specific use case or application. Selected attributes of Oracle's implementation include:

[4] For a complete list of all the new database features and enhancements, including SQL macros, AutoML, graph improvements, and more, readers can visit Oracle's blog introducing Oracle 21c: *Introducing Oracle Database 21c.*

- Immutable Table – This prevents forbidden changes by an insider with valid credentials by disallowing all database interfaces that modify data. New data can be added, but existing data cannot be changed or deleted by anyone, even database administrators.

- Blockchain Table – This protects against forbidden changes attempted by bypassing database software. Blockchain tables are immutable tables that organize rows into several chains. Each row, except the first row in the chain, is chained to the previous row via a cryptographic digest or hash. The hash is automatically calculated on insert based on that row's data and the hash value of the previous row in the chain. Timestamps are also recorded for each row on insertion. Any modification to data in a blockchain table breaks the cryptographic chain because the hash value of the row will change and can be easily detected with a built-in procedure.

- Distributed Digest – Protects against a large-scale cover-up, where the entire database is replaced. To detect such a cover-up, Oracle enables schema owners to sign and distribute the cryptographic digest periodically for a blockchain table. The digest can't be used to infer the data in the table, but authorized users can use it to validate the chain and confirm their newly inserted data is present. The cryptodigest can be posted to an independent public store or blockchain, like Ethereum, or sent out by email, or made available via a REST API. A cover-up can easily be detected by comparing the previously published digests to the current table content.

- Data Signing – Prevents impersonators from falsifying data by allowing end-users to cryptographically sign new data that they insert with their private key that is never passed to the database. End-users register a digital certificate containing their public key that enables the database to validate their data signature.

In-memory Improvements

Oracle Database in-memory functionality enables organizations to benefit from running both a columnar model *and* row-based approach simultaneously. In 21c, there are several enhancements, including in-memory vector joins, hybrid columnar scans, and self-management of in-memory column stores. In-memory vector joins are now possible by leveraging SIMD hardware acceleration and pipelined execution. Operations such as hash joins are optimized by being broken into smaller operations that get passed to the vector processor, yielding performance improvements of up to 10x. In-memory hybrid columnar scans are now possible, enabling customers to run queries that need data from the column store and row store. Lastly, the in-memory column store is now self-managed, meaning Oracle automatically manages the placement and removal of objects in the in-memory column store through usage pattern tracking.

In-database Machine Learning

Oracle Database 21c enhances in-database machine learning capabilities and facilitates the use of machine learning by users beyond data scientists. Featuring a new AutoML easy-to-use interface, non-experts need only "point-and click" to choose a data set, choose the data to predict, and press start: AutoML then automatically builds and compares machine learning models and provides a ranked list of algorithms based on prediction accuracy. Oracle has also added new algorithms for anomaly detection, regression, and deep learning analysis to its extensive library of popular, in-database machine learning algorithms. This is a superior approach to extracting data, moving it to compute, and using sophisticated tools outside the database to build and run machine learning models.

Persistent Memory Support

21c stores database data and redo logs in local persistent memory (PMem), which significantly improves the performance of I/O-bound workloads. SQL runs directly on data stored in the direct-mapped persistent memory file system, eliminating

the I/O code path and the need for a large buffer cache. In addition, new database algorithms prevent partial or inconsistent stores to persistent memory.

The benefits of these innovations will no doubt prove attractive to customers on Oracle's latest long-term release, 19c, or even older revs. There are many users who would want to take advantage of Blockchain Tables, AutoML, SQL macros, persistent memory, and so on. While Oracle has back-ported some of the key 21c features into a 19c release update, those on older revs will have to upgrade to 19c first. Another option is to migrate to Oracle Autonomous Database, notably for those customers looking to gain cloud economics in a 100% self-driving cloud database.

Alternate Approaches: A Note on the Competition[5]

Oracle's all-in-one converged ecosystem approach distinctly adds value when contrasted with its prime cloud database competitors. Alternative approaches fall into two main camps: either users can get most of what they need by investing in numerous databases, APIs, and management tools *or* they can make do with less than they really need by settling for a single offering that doesn't provide the prerequisite range and flexibility. Both are clearly sub-optimal tradeoffs.

Sometimes you will hear the competitive claim that Oracle only has a "hammer" and so everything looks like a nail. This is clearly erroneous; if anything, Oracle has a demonstrably extensive range of tools in one—and indeed one conveniently autonomous—toolbox.

An alternative suggestion is that users don't really need anything beyond a one-dimensional solution (even if that's true today, the odds of it changing are high, which means it is akin to deploying hope as a strategy). Indeed, this statement is also disingenuous, as we know that users are heavily focused on, and investing in, improving secure and collaborative productivity[6]…and one very clear path to that is convergence and interoperability across data models and workloads.

Oracle's chosen analogy is that either of these competitive approaches is much like choosing to operate an electric grid, compared to its own "simplicity-itself" approach of turning on a light switch. Why, says Oracle, bother with DIY, when you can have an integrated offering? And these crucial conceptual differences of approach between Oracle and its competitors are all before accounting for the more pragmatic "speeds, feeds, and specs" advantages that Oracle has in areas such as scalability (not only the physical size potential but also dynamic and non-disruptive auto-scaling), replication, sharding (sometimes referred to as shared-nothing horizontal partitioning), high availability, dynamic flexibility, and security.[7]

Real World Impact[8]

The impact for customers of examining and optimizing their database provision can be dramatic, both operationally and financially. Above and beyond all else (ignoring for now the respective practical capabilities of the various approaches in terms of performance and function), the crucial challenge for common alternative solutions to Oracle boils down to low resource utilization due to the over-isolation of databases; this drives database instance and data sprawl and leads inevitably to higher costs. An Oracle Database engineering expert shared examples with ESG of over-isolated databases on

[5] For more extensive competitive analysis, see also the ESG Showcase *Extracting Maximum Value from Your Data with Oracle Autonomous Data Warehouse,* published December 2020, and available on Oracle's website.

[6] Multiple insights from ESG's Research Report, *2021 IT Spending Intentions Survey* show collaboration, security, and productivity at the top of IT initiatives/drivers.

[7] Again, the ESG Showcase, *Extracting Maximum Value from Your Data with Oracle Autonomous Data Warehouse*, published December 2020 (available on Oracle's website), has more detail on how these comparative issues play out in the real world.

[8] This section is drawn from an interview conducted by the author (MP) with an expert in Oracle Database engineering. Although the specific client details quoted have not been directly verified by ESG, ESG was shown calculations, user data, and outcomes that are summarized. The referenced expert has over 20 years of experience with Oracle Database consolidation and capacity planning projects.

multiple platforms (such as VMware and IBM pSeries). It was no surprise (given Amazon's considerable strength in the market) that the main comparison he wished to highlight was that of a large customer using EC2.

In the example, the customer had provisioned over 1,500 processor cores (3,000 vCPU) across more than 500 EC2 instances but was using only 31% of the vCPU resources at peak (only 18% concurrent utilization).[9] Tighter management to drive up utilization was not really feasible at this client, since some of its EC2 instances had only 1 or 2 processor cores and it was not possible to reduce their size. Using its Exadata Cloud Service, Oracle recommended consolidation—based around 2-4 database cohorts in this case—to get the user closer to its potential 18% utilization. The financial implications[10] can be significant; using list pricing, the customer could save around two-thirds of the AWS cost, with the added bonus of higher performance and much less administrative effort for such things as upgrades and patching.

The Low-code Episode: APEX to the Future

Oracle APEX is a low-code development platform that enables developers to rapidly build data-rich applications for the enterprise. Available for well over a decade with Oracle Database, APEX has a trusted following of more than 50,000 customers and a community of over 500,000 developers and partners that produce thousands of applications a day.[11] In a world where speed, agility, efficiency, and quality matter (often more than many other business objectives), APEX is a platform that can enable everyone from advanced developers to those with no development experience to build applications that matter to them, to their jobs, and ultimately to their business. Oracle continues to look for ways to enable the next generation of developers as well as citizen developers to rapidly build transformative, data-centric applications. In fact, Oracle's mentality is that while traditional coding will still be relevant in certain areas of the business, traditional coding should be an exception and low-code should be the rule. Through Oracle's own testing as well as customer testimonials, APEX has proven to speed up application development by as much as 38x when compared with traditional coding; and with ESG research showing that 44% of organizations plan to increase their spending on application development and DevOps tools over the next 12 months,[12] APEX will surely help land Oracle on many shortlists.

44% of organizations plan to increase their spending on application development and DevOps tools over the next 12 months.

APEX delivers more than just application development. It enables applications' lifecycle management. While it starts with an application builder, end-users can then create tables, model data, and apply business logic. When applications are ready, they can easily be deployed with a single click and RESTful web services are easily publishable to go along with them. APEX also offers the ability to collaborate with other developers within the organization throughout the application lifecycle. Put it all together and APEX delivers a complete low-code environment to quickly build, manage, and iterate on transformational web applications. Moreover, it's available anywhere that the Oracle Database runs, whether that is on-premises, in Oracle Cloud Infrastructure (OCI), or any third-party cloud that supports Oracle Database.

[9] Utilization rates in this user's RDS population were very similar.
[10] Oracle's Cloud pricing model includes a fixed price component to reserve capacity (Exadata X8M compute and storages), and a variable component (OCPU) that can be set to the level of utilization.
[11] Numbers quoted by Oracle in its announcement event.
[12] Source: ESG Research Report, *2021 Technology Spending Intentions Survey*, January 2021.

Oracle's APEX Application Development Service

With APEX Application Development Service, Oracle is capitalizing on the growing interest in low-code development experiences and leveraging the maturity of its own APEX platform. It is a fully managed service running on OCI that wraps the core APEX platform around/with Oracle Autonomous Database and Exadata Cloud Service.

Fast On-ramp

APEX can be launched directly from the OCI console to provide users with the ability to quickly create new instances, manage existing instances, or start developing an application with a single click. And that's just the appetizer in this "easy button" meal. The underlying service is an all-inclusive package of "low-code-developing-goodness," where developers can utilize Oracle REST Data Services (ORDS) to create and manage REST endpoints and SQL Developer Web to model data and write SQL queries using a graphical interface. The service provides developers with the elasticity of a cloud environment, ensuring the dynamic availability of the right level of resources. And if application developers and managers feel an application has evolved to require the more robust Oracle Net Services, they can upgrade the application with a single click as well.

Fully Managed

Having the APEX Application Development Service built on top of Oracle Autonomous Database (ADB) means that all the benefits of ADB naturally extend to the APEX service. That means, for example, that the pre-configured and pre-tuned underlying Database, its resources (i.e., storage, compute, networking, etc.), and its security policies are inherited. And, while the underlying infrastructure and software capabilities are patched and upgraded automatically, so too then is the APEX service. This includes the addition of new low-code capabilities and APEX functionality twice per year. While these upgrades can be deferred for a set period, they have yet to require any changes to the applications themselves. In other words, for users, this represents the lowest of low responsibility bars when it comes to managing both the APEX service and its supporting software/hardware stack, which in turn of course means users can keep more focus on creating applications that matter instead of "administrivia."

Low Cost[13]

Starting at $358 per month, customers can use the APEX service with 2 vCPUs and 1 TB of storage. Additional vCPUs are billed hourly at $0.1613 per vCPU. Additional storage is billed monthly at $118 per TB. There are no added costs for more users, developers, applications, or data transfers in and out of the APEX service. Moreover, developers can start building low-code applications at no charge using the new APEX service.[14] To put this in perspective, some competing application development platform services in competing clouds impose end-user limitations and, in some cases, cost as much as $10,000 per month.

The Bigger Truth

As the title of this paper suggests, these Oracle capabilities are best viewed in combination; they are the instruments within Oracle's big picture "database ecosystem" approach as it delivers its "magnum opus." From data types to development and delivery, Oracle's approach manages to combine integration, uniformity, flexibility, ease of use, and economy.

Clearly, Oracle is full-steam ahead on its converged database strategy, a marked difference from the multiple specialized, isolated, single-use databases that create silos and make data sharing difficult and costly, and which are typically proffered

[13] Pricing as of the time of writing in January 2022.
[14] This is included with *Oracle Cloud Free Tier.*

by its competitors (chief among these, of course, being AWS, MongoDB, Snowflake, and so on). Oracle's converged approach serves to amplify the value of organizations' data—for instance, enabling users to run inference directly on their database, right next to their data, while precluding sprawling APIs, ETL approaches, training needs, and data integration processes—all of which add time and cost.

Just as important as the database convergence is Oracle's embrace of a huge range of "consumption divergence." On-premises, in the cloud, cloud on-premises with architectural equivalency, flexibly and dynamically sized components, and each customer gets its pick of manual, automated, or autonomous deployments. While the range of choices can get semantically befuddling, the essence is simple: When it comes to Oracle Database, users can have precisely what they want, where and how they choose to meet business needs.

APEX is a hidden gem in Oracle's product toolbox even though it has a long history. Nowadays, it is clearly targeting the "citizen developer" both by being inexpensive (even with a free tier) and available separately from the database itself. Putting its clear value to one side, it is aimed squarely at "classy but cool" modern application developers.

Referring to the analytical parsing mentioned at the start of this paper, the improvements offered by Oracle Database 21c and APEX are clear; but what about the integration? Basically, this is all about getting the most out of users' databases—and their associated data and applications, of course—in as efficient and effective a way as possible.

The talented conductor concept introduced in the opening conveniently allows us to mention Oracle Development's database leader—Andy Mendelsohn. His last name is of course all-but-identical (albeit with the addition of an extra "s") to that of Mendelssohn, the famous 19th century composer. And Oracle's overall approach can indeed be viewed as that of a full-blown "symphonic database," with all the instruments of the "data orchestra" (models, workloads, tenancy) working in unison under the direction of its master conductor. Oracle is not focused on solo instrument sonatas, or concertos, quartets, or whatever—it is instead creating an orchestral database environment, one where even you and I can pick up the APEX baton and create great, new, varied but integrated database music in a highly productive way. Is this magnum opus finished? Of course not, but the movements and key themes are crystal clear, and will be music to the ears of many.

Enterprise Strategy Group is an IT analyst, research, validation, and strategy firm that provides market intelligence and actionable insight to the global IT community.

www.esg-global.com contact@esg-global.com 508.482.0188