

Distributed Caching: Why It Matters For Predictable Scalability on the Web, and Where It's Proving Its Value

What about the Web is predictable today? Organizations continue to add more users for their Internet applications; more ways for these users to access their sites; and a lot more content, of a much richer nature.

To build customer satisfaction and loyalty, there are more requirements for businesses to let users participate without disruption in everything from online surveys to shopping across their discrete brands, and there's more pressure to let them speedily update everything from profiles to purchases.

That all contributes to a lot of unpredictability: an ever increasing number of calls to databases and meta data stores; continuously high incidences of data writes by many thousands of users; and a rising necessity to share more users' session data between different Web applications, domains or application servers. In a world of consistently spiraling data demand and submissions, and only so much capacity to actually supply and accommodate it, pity the poor IT organizations — yours likely among them — trying to scale lower-tier data-

InfoWorld
Custom Solutions Group

bases and back-end operational data stores to keep up with the expanding client tier. Given the potentially viral nature of Web applications, IT can only guess at growth surges, and guessing is a dicey game when it involves risks like damage to your brand and reputation, loss of competitive advantage and ultimately loss of business. And that's not to mention the expensive software licenses and large hardware system purchases typically associated with adding database capacity.

Some degree of uncertainty comes with the Internet territory, of course. No enterprise wants to limit the number of customers that can sign up for its Web-based services, for example, or the number of transactions they can conduct online. Nor would it want to sacrifice the ability to add new features or functionality for fear of breaking the site or losing competitive advantage that regular refreshes are actually intended to increase. The problem comes when the costs of adding additional capacity to handle growing loads, without incurring performance and availability penalties, climbs at an unforeseeable rate — and a rate that ultimately could become unsustainable, too. That's the kind of unpredictability no organization can afford.

What's needed to resolve the tensions that exist between the upper client tier and the lower tiers in the stack is a broker for data demands. Such a technology creates predictable scalability in today's frenetic Web environment, where it's impossible to know in advance how applications will need to scale — and so impractical to rely on the traditional ways of architecting systems to pre-defined metrics for users, transactions and growth. What's needed is to move away from the methodology of having applications query the database directly each time data is required to be retrieved, updated or passed around, to one where data lives closer to the application tier, the nexus of data demand.

ADDING COHERENCE TO YOUR INFRASTRUCTURE

This is where Oracle Coherence Data Grid technology enters the picture — where its innovations in distributed caching make their mark. Developers are familiar with the idea of writing applications to temporarily store data in local memory on the machine on which a workload is running, as well as with the limitations of that model. Based on memory consumed by the operating system, by Java for a Java Virtual Machine

and by the application server, the residual amount of memory available for storing temporary data can be quite small and the possibilities for resource contention that slows down performance quite significant. One can add more application servers to address server memory availability issues within an application server cluster, but that adds to operational cost and complexity, and may actually lead to additional memory consumption issues.

But adapt the local caching idea so that now you're combining memory across multiple systems in a data grid independent of the application server cluster, and the door to predictable scalability opens wide. You create a large and expandable memory footprint for reliably managing data for the application tier, and it is finally possible to cost-efficiently solve the challenges created by the enormous thirst for and pushing of data that originates at the client tier.

Now data can be moved off the back-end data sources and stored in memory in an expandable on-demand distributed caching tier where it can be made available to different applications as needed, and also where it can be offloaded to avoid lag times for transient storage needs. Instead of reaching out to the limited amount of memory on a local machine or to the back-end database source, this new distributed caching tier becomes the mechanism for predictable scalability. When you don't have to increase the load on the back-end data source, you don't have to scale it up. When you put Coherence in front of that database to offload repetitive reads and writes, you've contributed to winning back considerable capacity as well as increased system productivity — ultimately increasing the database's life span. That's a strategy that appeals to many IT leaders, including CTO Stefan Piesche of email marketing Software-as-a-Service vendor Constant Contact. "One of my big missions is to move the company away from scaling up to scaling out," says Piesche. "We're looking for alternative strategies to deal with excessive load and use cases that are hard to scale at the database tier itself."

Predictable scalability comes at a predictable price. The cost to increase the capacity of the distributed caching system is a known quantity: It is the price of adding a commodity blade server, or any commodity server, with a lot of memory and a Coherence license, with the operating costs for that (additional electricity consumption, for example). That's some-

The Future of Online Legal Research: **Coherence Plays a Big Role**

A leading name in the online legal research space is looking to take its services to the next generation. The company has set ambitious goals for the latest incarnation of its service, chief among them ensuring a seamless customer experience by leveraging its highly modular and scalable architecture. Customer satisfaction is a priority as the service's adoption rate rises faster than anticipated. The company now has expectations that its latest service, launched in March, will support 15,000 concurrent users by year's end, up from its original predictions of 10,000.

Oracle Coherence is a critical underpinning of the entire application, assuring that the service can be delivered as designed to meet the expectations that lawyers and other information professionals have for swift access to their credentials, preferences and search results when they return to the service after logging out. Fast access helps users be as productive as possible while keeping costs down. The company serves up the service through three different data centers, which each maintain multiple instances of the online legal research solution, and has a strategic road map of deploying more facilities globally. Coherence's in-memory data grid technology steps in to store the workspace a user has created so that it is shareable across all the instances in the data center grids. So, when a lawyer returns to her desktop after a short time away — say, after presenting a case in court — she can immediately pick up where she left off, regardless of which instance she connects into.

The company also has considered the case of users who might need to access their workspaces after a much longer time away — weeks even. They'll want to be able to retrieve their credentials and results with equal speed. To facilitate that, it has tightly coupled Coherence and the Oracle Berkeley DB, so that when a Coherence cache fills up, older search results can move to the latter, which functions as a near-cache. That lets the service provider give users the capabilities they need without incurring the expense of going to the underlying database.

Another critical customer service that the company is considering is using Coherence in conjunction with auto-balancing across instances. In that case, Coherence could be used as a medium for transporting user credentials to another module of the service; that way, a user could continue working without disruption or re-authenticating when existing module resources are low.

The company saw its Oracle Coherence deployment go off without a hiccup. In-memory data grid technology is powerful but, unlike databases, there's more to these deployments than just set up and go. Much of the credit for the success goes to a two-way partnership between its in-house Coherence gurus and Oracle's own experts. Milliseconds translate to significant dollars spent or saved for the online legal research services provider, and thanks to working with Oracle, the company has been able to realize its performance goals for Coherence.

thing an IT organization can lay out in plain English to line of business leaders. "Coherence really has allowed us to be very predictable in regards to how many resources we need," says Piesche, who implemented the Coherence*Web module. "We can add blade servers fairly transparently and that allows us really to become very predictable — to say that if we want to support another 50,000 customers, this is what we will do."

There's the added bonus that distributed caching reduces not only the cost but also the risk as compared to scaling up lower tiers: With Coherence, a new resource can be added on the fly to a production cluster. Nothing needs to be taken down during the process.

PREDICTABLE SCALABILITY PLUS

The main advantage of predictable scalability is accompanied by key supporting benefits, including better application responsiveness and increased flexibility. When frequently used data is closer to the application tier and set free from contention for machine resources, access to it is faster. And because data in a Coherence cluster is stored on both primary and backup servers, even the loss of a machine in the cluster doesn't result in any disruption to operation of the application or, more importantly, the loss of data. Data is continuously available even if a server fails. When you scale out the technology, because there are so many resources in a cluster, system failure isn't a catastrophe. Oracle Coherence automatically rebalances data

without requiring any human intervention, always keeping data and the cluster in a reliable state.

Fast access and ensured availability are both important to improved application responsiveness for online businesses that connect to their customers. Availability certainly was important to the development of an online legal research service from a leading provider in the industry, which uses Oracle Coherence in combination with Oracle Streams to share a cache across cluster nodes, so that another node picks up if one fails. “The first time we turned on the scale test, to everyone’s delight it scaled up quickly to many thousands of users. The architecture and all of the high-availability elements functioned perfectly,” says a vice president and chief architect for the organization.

Flexibility results because the cost controls afforded by predictable scalability and the advantages that enable application responsiveness enable an IT organization to be fearless about debuting new capabilities. It can deploy new functionalities to keep Web sites fresh, attracting and retaining customers without concern that it’ll be creating loads that the system is unable to support from the standpoints of efficiency, cost-effectiveness and reliability.

WHERE PREDICTABLE SCALABILITY MATTERS

The distributed caching capabilities that drive predictable scalability enable organizations now to effectively and with economically serve three primary “data demand” use cases for Internet applications:

- Repetitive reads;
- Repetitive writes; and
- Session state management.

▶▶ **The first scenario probably presents the largest use case for Oracle Coherence distributed caching technology.** For many Web applications — online catalogues, for example — end users around the country are likely to access the same data many, many times over the course of a day. Wherever those shoppers are located, and however many of them there might be at one time, they all want fast access to item prices and other information. Slow performance tied to data access could very easily send shoppers used to immediate satisfaction off to some other Web site. Making that data available in a Coherence cache solves

“The users on the server just move over to a different server, recover the session there from Coherence, and it’s so fast they don’t even know it’s happening.”

— STEFAN PIESCHE
CTO, Constant Contact

that problem. (Users are guaranteed the latest data because information is updated into the cache when any alterations take place; additionally, options such as invalidation strategies give organizations the opportunity to expire particular data sets in the cache every x minutes and retrieve new information from the back-end database.)

Executives at one Web site that is well known for, helping consumers get information on automobile research and new car inventory recognize there’s an adverse impact on revenue when page load slowly and access to information is frustrated. “We see people coming back to the site less, we see people submitting less leads, we see people viewing or consuming less pages,” says an executive director of software architecture there. When planning its redesign to hit its business goals, the company decided to get very aggressive with page loads — 75 ms. for the first byte to come down to users and 1.5 seconds for them to have a functional Web page with which to interact, he explains. Coherence is becoming the primary data source for powering this Web site, playing a big role in the plans to provide pages very quickly.

“There is no product out there that does what Oracle Coherence does,” the director says. “It has no points of failure, it allows me to distribute computation to any member of the data grid, and it stores an enormous amount of data and delivers it very, very quickly to our end users.”

▶▶ **Data updates present the second challenge for**

Web applications: How is it possible to let end users individually update what may equate to millions of rows of data in a back-end database without sinking the system? You may have a very fast and very optimized database server, but greater data volume and an increasing number of transactions will at some point take their toll on service level objectives. Databases are built to handle inserting big chunks of data in blocks, and a Coherence cache plays to that strength. Use it to offload batch writes of transient data, such as shopping cart stores or online gaming transactions, to the database at specific intervals. The Web application will perform better, as users don't have to wait for data to be written before they can move along. And you can add more users to the site and let them create all the work they want without dramatically increasing the load on the back-end database, thanks to capabilities such as bundling multiple changes to the same object so they're written only once.

▶▶ **The session state management issue is certainly one where the rubber hits the road for many Web application users.**

Their interactions with your site over the course of a session — say, filling out a registration form — won't be viewed in a favorable light if five minutes into inputting data the application server crashes and their session that was tied to it is lost. That won't happen if Coherence*Web — is deployed in the infrastructure, as the objects within the session state can be cached, and thereby live on even if the application server goes offline for unplanned downtime or even planned maintenance.

At the same time, a corporation's distinct brands may fail to grasp opportunities to boost stickiness and cross-sales potential without a way to let customers traverse siloed infrastructures and shop at will. Coherence*Web enables this as well, by holding within its cache session information and then migrating that data between sites as necessary.

THE LEADERSHIP FACTOR

It is possible to achieve distributed caching with other platforms, of course, both commercial and open source. Compared to other commercial offerings, however, Oracle Coherence is the most mature and most widely adopted solution in the market. Oracle's continuing innovations on the technology include integrating it as a part of its WebLogic Suite 11g,

to smooth the way for customers to leverage the WebLogic application server in conjunction with Coherence. Gartner positioned Oracle in the Leaders Quadrant in its 2009 report "Magic Quadrant for Enterprise Application Servers"¹. That said, another strength of Coherence is its seamless integration with application servers from other vendors. Constant Contact, for example, has deployed it successfully in conjunction with the JBoss Application Server.

Coherence is often measured against open source caching solutions IT leaders can deploy at no upfront cost. The latter may well present as a very attractive alternative to adding more licensing costs to IT budgets. But implementing a distributed caching solution isn't a trivial business, regardless of the solution. When organizations opt for an open source offering, they're pinning their hopes on the idea that whatever challenges they face in their particular deployment will have been faced — and solved — by someone else using the same solution. That can be risky for businesses whose online operations represent a heavy percentage — perhaps even all — of their revenue base.

Clearly, support was an important factor for Shopzilla, which deployed Coherence as part of its infrastructure. Answering a question about the use of open source alternatives in response to a blog post he wrote on Shopzilla's use of Coherence, Rob Roland of the online comparison shopping site noted that "none could match the response time of Oracle's support staff for issues."

He also noted that "none of the open source alternatives were as feature rich as Coherence," after enumerating the value of its "caching semantics, including automatic redistribution and partitioning of data when nodes are added or removed. It has built-in support for implementing 'Cache-Stores' that can perform read-through for cache misses and write-through or write-behind to any data source for backup. The query functionality (tied to their powerful serialization implementation) offers fast, compelling query access, more than just a simple 'get' of a key in the map."

His comments are echoed by another Shopzilla executive who noted that, while there are several viable open source alterna-

¹ SOURCE: Gartner Inc., "Magic Quadrant for Enterprise Application Servers - Yefim V. Natis, Massimo Pezzini, Kimihiko Iijima - 24 September 2009"

tives, their lack of out-of-the-box capabilities that Shopzilla needs for operating as a large, mature marketplace would have meant spending time “having to develop a significant engineering competency in the ...solution itself, rather than focus[ing] on delivery of our core value proposition (shopping). In our case, Coherence did what we needed. It isn’t free, but given the trade between dollars and engineers, we were fortunate to be able to choose dollars this time.”

That is not to say that implementing Coherence is free of any development requirements. The Coherence*Web session state management module is designed to be plugged in without requiring code changes, but other use cases require adaptation of applications. That will take third-party software such as Coherence out of the mix for repetitive read and write caching scenarios, but many large organizations tend to write their own Web applications anyway. To that end, it’s worth noting that Coherence uses the same Java API that developers know well, so it’s very easy for them to quickly become productive with Oracle’s distributed caching technology.

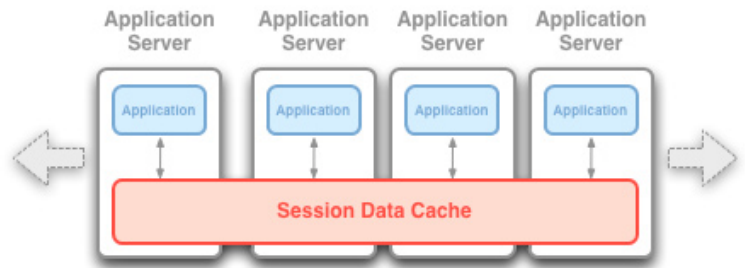
COHERENCE HELPS CONSTANT CONTACT

Coherence*Web, as it happens, has become an important caching tool in Constant Contact’s tool belt. The SaaS provider that helps small- and mid-size businesses with their email- and event-marketing and online survey needs knows very well the challenge of being able to support spikes in customer demand without disrupting the customer experience. Some 350,000 companies take advantage of its services, particularly its data-intensive tools around email marketing campaigns for preparing and editing messages to their own clients, and how often each one uses its solutions and exactly how many customers are concurrently in the system is always in flux. One thing that doesn’t change, though, explains CTO Piesche, is that a lot of content is moving around as those customers engage in their email editing processes, with a lot of documents being continuously updated.

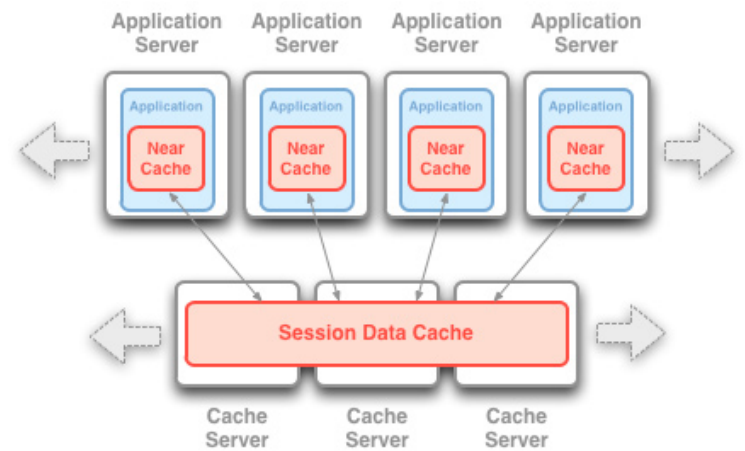
Now imagine those users one-quarter or halfway through their email marketing project — or perhaps putting the finishing touches on it — when the session is lost as the result of an outage. “The email editing process is very data intensive — you’re editing the email template and manipulating a lot of data within a session, adding styles and referencing

COHERENCE*WEB SESSION

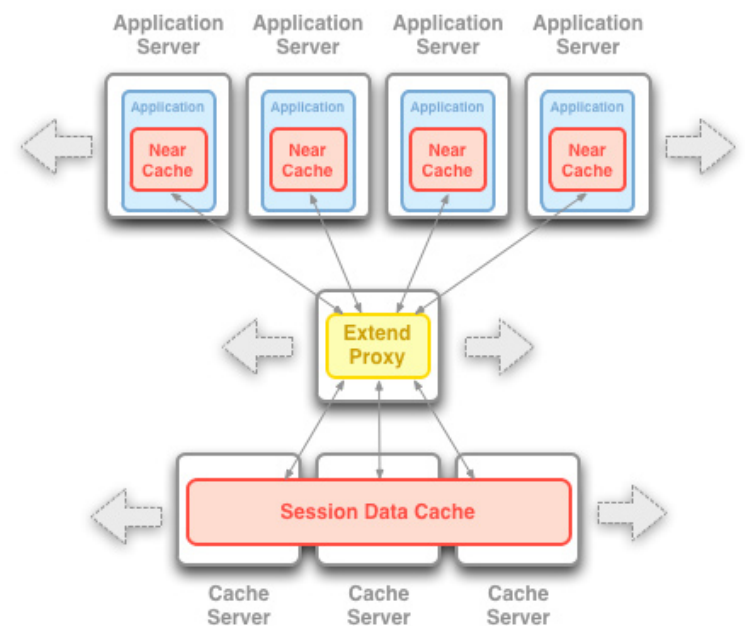
In-Process Deployment Topology



Out of Process Deployment Topology



Out-of-Process with Coherence*Extend Deployment Topology



images — all sorts of things, and those editing sessions can get very large,” says Piesche. It isn’t unrealistic to expect as many as 30,000 customers to be engaging in the process simultaneously, working on newsletters that could be 1 or 1.5-megabytes each. That’s on the order of 40 gigabytes of workflow-related data to be maintained on the fly and provided in a low-latency fashion. It’s clear how important it is for such large sessions to be handled efficiently so that users feel as if they are working on their own desktops, and equally clear is the requirement for fault-tolerance, so that the large and transient data sets added within a user session survive a software or hardware server failure.

Not long ago Constant Contact was living with the fear that such an event would jeopardize the user experience and customer satisfaction. “They would lose some of their urgently being edited data, because the server went down and they hadn’t saved or it hadn’t auto-saved yet,” Piesche says. “Or even if they didn’t lose data they would have to log back in, click through to where they were, and it takes about a minute or two to get back to where you left off.... Our customers don’t have a lot of time to spend on email marketing, so wasting any of their time is really bad.” Data loads were simply too high to use the normal replication mechanisms that application servers provide to address the issue, says Piesche. As soon as he joined the company a year ago, he was determined to solve the problem, which led to the decision to bring on board Oracle Coherence*Web.

Since Coherence went live a few months ago, Piesche no longer worries about outages. Coherence lets session states be managed in a variety of caching topologies and enables session data to be stored outside of Java EE application servers. That means that application server heap space is freed up and servers can restart without session data loss. “The session moves to a different server and customers never know anything happened, so that helps us with uptime stats from the point of view of customers,” the CTO says.

It isn’t just those unplanned events that can disrupt the day-to-day user experience where Coherence*Web’s ability to manage user sessions in a cluster of production servers comes in handy. It also helps Constant Contact gain an advantage in everyday planned infrastructure maintenance and software deployment. With Coherence*Web installed,

the SaaS vendor can take down a server to install a patch or immediately upgrade a system with the latest version of its software without impacting the work a customer has in session. Being able to be as efficient as possible with upgrades matters a lot in the SaaS world, where it’s important to rapidly deploy new functionality to keep customers eager about and invested in using a provider’s services. “The users on the server just move over to a different server, recover the session there from Coherence, and it’s so fast they don’t even know it’s happening,” Piesche notes. And now there’s no need for Constant Contact to continue to maintain a second farm of application servers just so that customers can continue their work-in-progress while the company gets its latest deployments under way.

Coherence’s ability to hold Websphere and JBoss application server sessions in the same cache was also instrumental to the SaaS provider’s rollover to JBoss from Websphere. Without it, Constant Contact would have had a difficult time smoothing out the inevitable gotchas that accompanied its JBoss rollout (what deployment is ever free from those?) — customers on that platform were just passed over to a Websphere server as necessary while issues were resolved. “The specific feature to support a mixed environment was key for our successful middleware migration strategy,” Piesche says.

HARNESSING THE OPPORTUNITIES

Indeed, Piesche looks at the deployment of distributed caching technologies as critical to his mission as CTO: Preparing the company for growth and being able to support a future that includes scaling for more customers and more products.

Some IT leaders may look hardest at the risk of deploying a disruptive technology such as distributed caching, but their eyes should be on the rewards. “Taking these types of small calculated risks — what we get in return, like the peace of mind to update software on the fly without impacting customers, to remove servers without upsetting the customer experience — that is worth every penny,” Piesche says.

Other enterprises come to a similar conclusion when the opportunity to make a change looms large, such as when a merger requires a refresh of the IT architecture or a compelling event disrupts the business so much it becomes clear that traditional ways of solving problems aren’t viable. For

Coherence Also Counts For SOA and Cloud Computing

Oracle Coherence matters for two important 21st century IT trends: Service-oriented architectures (SOA) and cloud computing.

- ▶▶ **Why SOA:** SOA is effectively putting a Web service in front of a data resource so that that resource can be commonly used by multiple applications. That lets more people access data but also opens up the possibility of abuse by over-access. With Coherence, it's possible to cache frequent repetitive Web services requests. Instead of executing that service by going to the back-end data source, the data is cached in Coherence and can be reused from there.
- ▶▶ **Why Cloud Computing:** The move to cloud computing demands that developers steer away from creating applications that run in single instances on individual servers. As the model gives way to developing and deploying inherently distributed applications that run as a single instance across dozens of servers in private cloud infrastructures hosting multiple applications, Coherence can serve as a data abstraction layer for those environments.

many online retailers, Cyber Monday is that event — when the load doubles in volume year over year, survival without deploying distributed caching is a seat-of-the-pants event. Enabling predictable scalability with Oracle Coherence is the only way to ensure that application loads can grow at 20 or 30 percent a year without causing infrastructure mayhem.

The online legal research service provider saw an opportunity to take advantage of Coherence as part of its plan to

revolutionize its offering. It has set ambitious goals for the latest incarnation. Chief among them is ensuring a seamless customer experience by leveraging its highly modular and scalable architecture. (See Siderbar)

CREATING YOUR FUTURE

Not only can you harness opportunities when you deploy Coherence, but you also can create them. One online travel site, for example, is able to cache data it pulls in from individual Web service reservation systems, such as hotel room information, into its Coherence cache so that the information is quickly available to users. That's good for consumers looking to book a room online, but it's even better for the site operators who, with knowledge of supply and demand, have the opportunity to increase prices based on availability and add to the site's margins. That's only possible because now it can perform the high-speed calculations necessary on data that it is able to hold in memory.

It's clear that Oracle Coherence will more than pay back the investment you'll make in the technology and in re-architecting applications to enable the predictable scalability it will deliver to your infrastructure. Offloading database demands of Internet-facing applications by 70 or even 80 percent is a huge benefit — one that significantly outweighs any costs around application changes. And based on the experiences of some leading online retail sites, an organization can implement Oracle Coherence in just a six- to nine-month timeframe.

Piesche sums up the advantages: "Coherence is such a robust product. Its failover behavior is excellent, and everything just works from an operational and reliability and scalability perspective," he says. "I think the best way to scale a database is by not going there anymore. Cache as much as you can in the middle tier." ||

About The Magic Quadrant

The Gartner Magic Quadrant is copyrighted 2009 by Gartner, Inc., and is reused with permission. The Magic Quadrant is a graphical representation of a marketplace at and for a specific time period. It depicts Gartner's analysis of how certain vendors measure against criteria for that marketplace, as defined by Gartner. Gartner does not endorse any vendor, product or service depicted in the Magic Quadrant, and does not advise technology users to select only those vendors placed in the "Leaders" quadrant. The Magic Quadrant is intended solely as a research tool, and is not meant to be a specific guide to action. Gartner disclaims all warranties, express or implied, with respect to this research, including any warranties of merchantability or fitness for a particular purpose.
