THE CLIPPER GROUP
Captain's Log™
Navigating Information Technology Horizons℠
Published Since 2001 Report #TCG2013023 November 18, 2013

The Search for an "All-in-One" Solution for Your Data Center — Why You May Want to Think Differently Now

Analyst: Jim Baker

Management Summary

Building a house used to be done one-at-a-time and all on site (ignoring prebuilt "mobile homes" destined for a trailer park). Even if you were building many of the same or similar houses, this was true. Each specialized crew (e.g., finish carpenters, electricians, and roofers) would move to a second house when their work was done on the first house. If each house was different, then someone had to worry about the many design intricacies that made it a solution for a specific kind of buyer. Think of this as "custom-built". Custom building almost always is more time-consuming and more expensive than something being built repetitively. For many buyers, this was the only way that they could get what they wanted. However, as time passed, many components and designs became standardized, to varying degrees; some now are fabricated offsite in a factory (rather than out in the elements). A good example of this at the component level is roof trusses (those typically-triangular shaped assemblies that hold up the roof). They always used to be built on site (a time-consuming, tricky affair) but today that rarely happens. Today, you even can buy a whole house as a kit, to be assembled on your lot, but most of us wouldn't want to do this. However, buying something preassembled has many advantages because it is ready-to-go soon after it is put into place and someone else is taking responsibility for its design worthiness and quality. **Preassembled infrastructure can make a lot of sense, if it can satisfy your requirements and fit within your budget.**

Fast forward to the data center today, where something similar has been happening with "multi-purpose IT infrastructure", made possible by virtualization of some or most of the resources (servers, storage, networks, etc.). It is multi-purpose in that it tends to run many applications, which may be independent or interrelated. Today's multi-purpose virtual infrastructure can be seen as a "virtual data-center-in-a-box", with many options and sizing possibilities. You might have only one preassembled solution in your data center or many, depending on your needs, how they scale, and how you want to acquire and utilize them.

You can assemble these units yourself (as has been done for decades) or order one (or more) units of them preassembled and grow by adding more. There are two big questions.

- *Do you want to assume responsibility for the design, integration, and testing yourself or pass those responsibilities on to someone else (a vendor or integrator)?*
- *Do you care whether these are architected in traditional ways or, maybe, do it differently?*

Of course, economic considerations and operational aspects are involved in your decision making.

IN THIS ISSUE

Today's scale-out data center often looks like it was built incrementally. There are many fine components and tools therein – and more than likely, there are many, probably too many, as will be discussed. We have many kinds of servers, storage, acceleration, optimization, communications, security, as well as monitoring and management devices or software, each often deemed to be "best of breed" at the time that they were procured, usually for a specific purpose or reason. Often, we see a similar collection (or set) of necessary devices in each rack, usually with many units of servers and storage. There are a lot of these amalgamated collections in the data center. Once upon a time, that collection (in one or more racks) might have come pre-bundled from a vendor and was sold as a "computer system". Then we tried to make each rack optimized for one or more specific purposes, a noble intention but the source of headaches for many. Now, we see some vendors again pre-bundling (and pre-assembling) the pieces into a computer system that can do many things and its capacities are sub-dividable by virtualization of many kinds. **Many vendors don't want you to focus on what is "under the hood", but simply to buy their "new-age building blocks" of "multi-purpose all-in-one infrastructure" designed for a virtualized data center.**

Many data center decision makers seem to prefer it that way. In many ways, it's like buying an in-house cloud, with all of the inherent arm's-length relationships (and detachments). With external cloud providers, in theory you don't have to worry about *what's* under the hood or even where it's located. **Buying, deploying, and using multi-purpose IT infrastructure, externally or internally, no doubt is easier when you don't spend much time looking under the hood (and tinkering with it, as well). Perhaps that's your IT goal – worry-free (less burdensome) infrastructure. But there is a caution: be careful to ask "at what price?"**

You know that this approach very likely is going to cost you more to let someone else worry about your problems, but that's why being free of worry is so attractive. Every organization must make this tradeoff decision for itself, by balancing its needs and what it costs to satisfy them. In the end, you need to know whether you can afford to do business this way and whether and in what ways it is beneficial or an improvement of prior practices. If you can, it may be the right thing to do. **Besides, it may not be all that much more expensive and indeed it may be worth the premium being charged**.

Looking Under the Hood

However, let's assume that you're just a little skeptical that you can be worry-free and still afford what preconfigured infrastructure[1] might cost. If you are concerned, then you must force yourself to look under the hood and ask, "Does this make sense?" Stated more pointedly:

- **Are the pre-configured "general-purpose" virtualized computer systems the answer to your needs?**
- **Is this the multi-purpose infrastructure that makes the most sense going forward for your business?**

To get started, let's take a closer look under the hood.

What do we find?

There are a lot of things that look like rack-mounted servers of various sizes and purposes. Digging a little deeper, we may find that there are a lot of *x86*-architecture servers (in the stack of components in the rack), with varying inclusions (storage devices, networking devices, acceleration devices, management devices, etc.). In order to make these into a multi-purpose device, all of these special-purpose servers need to be connected (networked) together and tested. **Integration and test are two big reasons that the task should be left to experts and not done by do-it-yourselfers.**

Depending on what you have acquired (from totally integrated solutions to much less so), you may have different management tools to manage each of these specialized parts (whether components like servers, storage, and networking devices), racks, or systems. For many, this might look pretty cluttered. Yes, each component, rack, and system is *multi-purpose* (in terms of what it can do) but it may be terribly inefficient to manage and operate, especially when the management is done differently at the component level. **And that, at the end of the day, is what you need to do well – to get the most out of your**

[1] Preconfigured does not necessarily mean that the system has been specified generically as "general purpose", i.e., without regard to specific needs, either of a common sort shared by many data centers or designed to your own requirements. The key point here is that ***someone else*** is making the decisions, delivering a solution, and standing behind what it has delivered.

infrastructure without being overwhelmed by the complexity of the pieces, individually and collectively – and, of course, without spending too much.

Why do some solutions have so many specialized components?

That may be the most important question. There is a Darwinian answer here: "Because they evolved that way!" **Over the five or six decades of commercial IT, it has been observed, over and over again, that the evolution of specialized solutions is giving way to more general-purposed or multi-purposed ones.**[2] Competition and the desire to deliver "more bang for the buck" has moved us from separately-priced, special-purpose components (like individual and separate network, security and storage controllers) to ones that are defined more by the collection of software running on them than the specifics of the underlying hardware platform(s)[3]. **Stated differently, if you could run all of your operating systems, middleware, applications, storage, networking, and management on a general-purpose server with its own storage, wouldn't that make a lot of sense, especially if your multiplicity of needs could be met and grow just by adding more of these building blocks? This is one reason why now may be the time to consider changing your data center paradigm.**

There is a further trend from pre-configured multi-purpose infrastructure (under the hood) built out of traditional (individually specialized) components to an "all-in-one architectural approach" where the software defines the purpose of one or more "universal components". Think about this broadly, because thinking narrowly about just one kind of infrastructure component (like storage) will cause you to miss the important points that follow. Read on to explore this further and to learn why now you may need to think differently.

First, The Answers

Just like the TV show and board game *Jeopardy*, here are the answers that you need to consider. We'll get to the question(s) in a moment. Many of these bullets are statements of newly discerned realities, presented for your consideration.

- **Hardware of many sorts is becoming increasingly transparent, especially in virtualized environments.**[4] While infrastructure hardware is a necessary ingredient and its procurement decisions do affect the total cost of ownership, today hardware tends not to be the exceptional (differentiating) component, especially for open systems environments. It's not that you don't need hardware – of course you do – it's that many folks may not want to pay a lot of attention to the details of the hardware running their application(s), sort of like you often do when you go to a public cloud for virtual server instances or remote storage. Because lines of business and users tend to be more focused on middleware and applications and the service levels being delivered, it often is systems software that differentiates seemingly similar hardware. Consider how systems hypervisors (such as *VMware*, *Hyper-V*, *KVM*, and others) allow commodity-like hardware (e.g., multi-core x86 processors) to be shared amongst many virtual users and their applications – without specific regard for the actual components under the hood.
- **Do-it-yourself integration of best-of-breed components from different suppliers is, at best, a luxury that may no longer be justifiable, and, at worst, may be a plague from which you may never recover.** Self-assembly and integration – even when done well – just gets you to the starting line. At its worst, it can negate some of the valuable engineering, testing, and integration that a single supplier can bring to the solution, because they are doing it over and over again for many customers.

---

[2] This especially is true for custom hardware that became firmware in custom chips that eventually became software running on standard servers, and often in the form of "appliances".

[3] By "platforms", I mean hardware, virtualization environment, operating environment, and necessary middleware.

[4] In an era of "cloud thinking", some would say that the infrastructure hardware (think servers, storage, networks, etc.) is becoming or has become "irrelevant", especially to the lines-of-business and external users. From their perspectives, it is about the qualities of service provided (with an emphasis on response time) and the reliability (and other measures of goodness) of the service(s) being delivered. **However, the hardware does matter to whoever is providing the infrastructure.** If you work in the data center, this includes you. The point being made above is that the underlying hardware infrastructure probably is less visible (and even may be totally transparent or invisible) to those outside of the data center. For a non-data center analogy of this, as a passenger on an airplane, you define your satisfaction in terms of your expectations and how well they have been met (and not on the details of the jet engines powering the aircraft). However, if you are an executive for the airline, you see the plane as infrastructure that needs to be functioning properly and properly used to satisfy goals for customer satisfaction, profitability, and safety.

- **Time is of the essence.** The payback for going the do-it-yourself route may never be realized but, meanwhile, much time and many resources most certainly will be consumed. Data center staff time may be your scarcest and/or most expensive resource. This resource should be what you are optimizing, not wasting on something that can be done, usually better, by others.
- **Application and data needs are changing; sometimes dynamically (possibly in a matter of minutes or hours).** The focus is no longer on the individual piece parts but on meeting business and users service level requirements (for the many apps and uses of data). Keeping up with the requirements is essential but, once again, it just gets you to the starting line (albeit every single day). The high degree of virtualization plus today's dynamic demands for applications and data are factors driving IT decision-makers to look at all-in-one solutions as a new, maybe better alternative to what they have done in the past.
- **Much of the application workload can be divided into two categories: virtualized apps residing on a whole or fractional core and larger apps requiring many cores** (the latter includes both scale-up (SMP) and scale-out scenarios). Maybe there are other architectural scenarios, but having too many just makes life more complicated. Getting the most out of fractionalized cores is a lot harder than assigning full cores, especially when the needs often are changing dynamically. This is true at both the hardware level and at the virtual image level.
- **Thus, the focus should be on finding a place to run your complex collection of mission- and business-critical virtualized applications**, without having to put together the pieces or, even better, without having to think about the underlying infrastructure, much or most of the time. Not having to worry (as much as before) about what's going on under the hood also is becoming increasingly attractive.
- **Complexity is the "new evil" (replacing overprovisioning).** It needs to be driven out because, in the long run, it tends to dominate the TCO equation, far more than hardware. This is your new "holy war". Here again, not having to worry about what's going on under the hood now seems to be even more attractive.
- **You want the simplicity of "running your apps in the cloud"** (i.e., not worrying at all about provisioning what is underneath) **in an environment that you can count on and control...all at a reasonable cost.** Note that while simplicity is the answer, as always you cannot ignore the costs for achieving it.
- **Standardized environments make a lot of sense, so long as they don't lock you in unwittingly, either to vendor or architecture.** If you feel compelled to look under the hood to see "what kind of standardized environment is in there", then you may be doomed from the start. You need to ***use*** what's under the hood and not let it ***define*** what is being done there or how it is being done. You need transparent homogeneity and not physical homogeneity, since infrastructure will continue to evolve and improve and you will need more of it, over time. Said differently, you need to be able to scale transparently when it is time to upgrade.
- **Today, for many businesses, the standardized environment is dominated by *VMware* and applications running thereon.** There are other hypervisor environments, but this is the big one right now, and may be the best place to start if you are considering a change for your data center.
- **Achieving all of the above answers (goals) is important** both for the data center and the business(es) of the enterprise. How well you do this will define your total cost of ownership and, thus, how it will impact your bottom line.

That's a lot of reality but, no doubt, the list could be longer. In fact, you might want to add to this list after reading the rest of this bulletin.

And, Now, The Question(s)

Before reading ahead, ask yourself "If those are the answers, what are the questions?" *Really, please do think about it and then proceed.*

There may be many variations on the formative question(s), but they all have a ring that goes something like this: **"What should I be doing to change my data center infrastructure for the better, both technically and economically?"** Variations and underlying questions might include:

- ***How can I make my data center more efficient; maybe more cloud-like?***

- ***How can I be more independent of the many hardware and middleware vendors?***
- ***How can I focus on what is truly important, like application service delivery?***
- ***How can I lower my total cost of ownership and thus the costs associated with individual applications?***
- ***What pitfalls are avoidable?***
- ***What advantages of an all-in-one solution await my user community? My development community? My management community? My bottom line?***

Add your own questions to this list. Then read on for some questions and answers that should help you understand why the focus in the data center may be changing.

Changing Your Focus

To begin, how could or should you be focusing differently?

- **Use a Wider/Maximum Focus** — All-in-one solution sets allow the data center staff to refocus their decision-making from spec sheet hardware and software feeds and speeds to how well the solution will service business applications and/or users. The decision point moves up the hardware and software stack to the *real work* of the organization. (Examples of real work include insurance companies managing risk and rates of return for their customers and investors, non-profits seeking contributions to fund their good works, utilities delivering power/telecommunications/water to rate payers with near perfect uptimes, etc.).
- **Be Aware That There is a Wide Range of All-in-One Solutions** — Solution sets are almost as varied as the number of vendors delivering them. Some vendors have single building blocks that can be aggregated together to deliver all-in-one functionality. Other vendors have pre-chosen individual products from preferred vendors but have done so using tight engineering, testing, and integration and then summing them all into individual models in a range of sizes and configurability. Those who have taken this integrative approach often utilize configuration variability (often presented as different models) to match the special needs of their customers. It is a merger of customization and standardization, like homes largely built in a factory out of common components that are well understood and have been tested with time. Still others prebuild one or more giant-servers-in-a-box and offer them with little or no custom tailoring. Here is the good news – regardless of which approach is employed, almost all of the infrastructure decision-making, amalgamation, integration, and testing has shifted from your team to theirs.

Advantages of All-in-One Infrastructures

What are the advantages of an all-in-one solution?

Because the range of all-in-one solutions is so broad, the best way to tell whether one alternative is superior to another is to evaluate whether and how well each solution lives up to the following list of advantages.

- **Pre-Engineering, Pre-Integration, Pre-Testing, and Pre-Configuration Done by Experts** — Design, integration, testing, and configuration validation are all done ahead of time at the factory by experts in their domains of excellence. Some vendors will do it all themselves, but others will work cooperatively with fellow suppliers, especially when there are multiple elements within the prepackaged solution set. The point is that you are not doing these tasks; they are.
- **Engineering** — Engineering teams from all supporting technologies should work together across corporate and/or vendor boundaries. Each should support the APIs (or Application Programming Interfaces) of the other. Coordination of release calendars is an indicator. Are all supported technologies on the same release calendar or does one or two always seem to be a quarter or two different? Inquire about management and coordination meetings, including goal setting for all involved.
- **Integration** — Integration for converged or all-in-one solutions is the responsibility of the provider, not the consumer nor the data center staff as has been the case since the 1960s. Putting all the pieces together before the product set is placed into service is the integrator's obligation. Better integrators will offer better solutions so it is in his best interest to outperform his competition. Your chosen integrator is or should be assuring compatibility, testing, upgradeability, sufficient support services, and that there are mutual support agreements in place with all technology contributors. Likewise, the good

integrator will know in advance how "next gen" products will improve the solution set and how necessary migrations will work to the data center's best advantage – on a case by case basis – including your case!!

- **Completeness of Testing** — Thoroughness of testing lends credence and confidence that the integration is indeed true and complete. "Testing" here includes vendor self-testing in its laboratories as well as user-level testing in the real world at customer locations (so called "beta testing").
- **Support** — A single vendor takes the onus of support for the entire solution set. If that vendor needs more in-depth engineering support, s/he has well-honed escalation paths to experts at supplier organizations. In short, there is one throat to choke, one escalation path, and inherent expertise across all hardware, software, and networking domains. For the data center, this is a very good thing because there is one organization or one lead organization with which to work.
- **Modernity and Growth Paths** — Modernity means procuring all modern/contemporary elements. However, this analysis needs to be expanded to include growth and expansion paths – over time. You need to consider will the support structure be available as new switches, processors, storage, etc. come on the scene? The practical answer is that it MUST. The data center will need some type of statement of support like "we will support what we sold you for *x* years, where *x* is defined as the period following the end-of-marketing offers" from suppliers – in order to assure a smooth upgrade future.
- **Operational Management Once App is Placed into Service** — The criteria here is how good the operational management is, how intuitive it is, and how easy it is to use. The UI (User Interface) should appear as a single pane of glass. For more complex decisions, the user should be able to drill deeper into easy-to-use-and-comprehend screenshot representations. Yet, s/he should be able to pop back up to the more basic layer.
- **Simplicity** — For the sake of simplicity, the converged solutions often have pre-configured models with sub-models as needed. Of course, all of these have been subject to the rigors of integration and testing mentioned above. The net result is that there are fewer models/SKUs (or model numbers) to order. The decision focus now is at the function level not at the device level. The payoff is convenience improved for the consumer and the data center, including its buying teams.
- **Quicker Time to Value** — The bigger payoff is a "shorter time to value" wherein the systems are functioning in real time with live production data within hours or days of delivery to your data center. Standardization is facilitated, because there are fewer configuration combinations allowed; thus, there are fewer startup/standup errors. The result is operational systems quicker doing the work of your organization, which is the reason for this kind of procurement in the first place.
- **Reduction in Start-Up Time and Professional Services** — Customer hand-holding can be significantly reduced or eliminated, especially on site. Installation and/or tuning services are much less likely to be needed. These too speed time to value for the solution.
- **Availability of Software Solutions** — Software providers at all levels, but primarily at the application level, know there is a ready market for their products for converged solutions similar to the *iPhone App Store* or the *Android Marketplace* in the smart phone segment. Markets move fast because they are self-defining and expanding on a daily basis. As a software vendor, these are all good reasons to code for a universal infrastructure/system. There is repeat-ability for the buyer and ditto for the seller.
- **Seamlessness and Transparency of Upgrades** — Innovation of underlying elements in the IT infrastructure is inevitable so the seamlessness of the upgrade process is a criterion for vendor and product selection. Ideally, the applications should not have to go offline for upgrades and there should be no need to migrate or move data. However, if data migration is necessary, it should be automatic and take data center staff out of the migration process as much as possible (or as much as is preferred).
- **System Elasticity — both expansion as well as contraction** — These solution sets should expand and contract autonomically, when components or capacity is removed or added or when workloads are added or subtracted. Performance and response time improvements are obvious advantages but so also are avoiding overpayment and buying too much technology too soon. Rather, procurements can be made only as they are needed so as to avoid overprovisioning for standing idle/waiting resources. All-in-one systems are exceptionally agile in that regard.

What are the cloud-like characteristics that you should seek in your all-in-one solution sets?

- **Maintenance items**, such as provisioning and automated backups, should be opaque to end users and simple for data center administrators. As more space or power is needed, the user is accommodated automatically and transparently within the rules and priorities of the organization.
- **Application catalogs** are self-service and installations are push-button easy for the end user. No hands-on support is required, most of the time.
- **System resources** are applied as application loads increase and subtracted as they decrease. Think about workloads at 10 a.m. vs. workloads at 10 p.m. The solution expands and contracts in real time and as needed.
- **Quantification of support** is in terms of SLAs (or Service Level Agreements). These include uptime requirements as well as how long it will take to return to normalcy in the unlikely event of system failure or degradation.

What are the pitfalls to avoid?

- **Simplistic Compatibility Matrices** — You want more than a paper-based list of items that the vendor claims will work together. You want to know that the combinations have been engineered to work together. You want to know if and how the APIs are being employed. You want to know that all components have been exhaustively tested in multiple combinations including load testing that matches your organization's particular use cases.

How can I get to a more efficient buying cycle?

- **Find a simpler and shorter procurement and provisioning process.** The buying cycle can be shortened when a lot of variability has been removed, often by focusing less closely on the specific characteristics of the hardware. Then "order-ability" is eased. New vendors as well as many solid "old names" familiar to all in our industry can offer these all-in-one solutions.

Finally, how might we lower our total cost of ownership (TCO)?

- **Pricing** — Pricing can be lower for all-in-one solution sets, because of commonality of parts, economies of scale on the part of the suppliers, and sheer volume, The vendor has room to either price lower or have increased flexibility during price negotiations. Either way, this is good for the data center's budget.
- **Obsolescence Less Likely** — With a single lead vendor, you will not have to deal with each subelement managed in terms of its own life cycle and escalating support costs. Rather, you will be dealing with the whole package. Smart vendors will define add-on modules that extend the infrastructure rather than telling users to swap out previous solutions in their entirety.

Conclusion: Lowered TCO is the Biggest Payoff of All

The old *Buy and Integrate IT Infrastructure Yourself Days* are gone. Most of the time, putting it together yourself is not a good use of the skills or time of the data center staff. Instead, why not focus them on the *application problems of the day*, not the *technology acquisition process of the day*. True all-in-one or converged solution sets have shifted the work from your organization to the providers. IT infrastructure caretakers (as a job function) are no longer necessary or are very much less necessary. We are moving from handcrafted one-of-a-kind solutions to generalized solutions that more than meet requirements of most situations, virtually all of the time. Highly skilled, bright data center employees now can focus on meaningful challenges that will affect the day-to-day native operations of your organization.

All-in-one solutions are, on average, less expensive to stand up in your data center than the more traditional methods. Try thinking differently! It might be the only way that the data center can do what is being asked of it.

About The Clipper Group, Inc.

The Clipper Group, Inc., now in its twenty-first year, is an independent publishing and consulting firm specializing in acquisition decisions and strategic advice regarding complex, enterprise-class information technologies. Our team of industry professionals averages more than 25 years of real-world experience. A team of staff consultants augments our capabilities, with significant experience across a broad spectrum of applications and environments.

➢ ***The Clipper Group can be reached at 781-235-0085 and found on the web at www.clipper.com.***

About the Author

***Jim Baker* is a Senior Contributing Analyst for The Clipper Group.** Mr. Baker specializes in storage hardware and software, the relationship between technology and its application in business, and emerging technological trends affecting users and vendors in the Information Technology community. He recently joined The Clipper Group after three decades in hardware and software product marketing and product management at companies as large as EMC and Honeywell Information Systems and as small as IPL Systems and Datapoint Corporation. He most recently served as Research Manager for Storage Software at IDC in Framingham, MA. Mr. Baker earned a Bachelor of Science in Marketing degree from the University of Illinois at Urbana-Champaign and an MBA, also from UIUC.

➢***Reach Jim Baker via e-mail at jim.baker@clipper.com or at 781-235-0085 Ext. 212. (Please dial "212" when you hear the automated attendant.)***

Regarding Trademarks and Service Marks

The Clipper Group Navigator, The Clipper Group Explorer, The Clipper Group Observer, The Clipper Group *Captain's Log*, The Clipper Group Voyager, Clipper Notes, *The Clipper Group Calculator*, and "clipper.com" are trademarks of The Clipper Group, Inc., and the clipper ship drawings, *"Navigating Information Technology Horizons"*, and *"teraproductivity"* are service marks of The Clipper Group, Inc. The Clipper Group, Inc., reserves all rights regarding its trademarks and service marks. All other trademarks, etc., belong to their respective owners.

Disclosures

Officers and/or employees of The Clipper Group may own as individuals, directly or indirectly, shares in one or more companies discussed in this bulletin. Company policy prohibits any officer or employee from holding more than one percent of the outstanding shares of any company covered by The Clipper Group. The Clipper Group, Inc. has no such equity holdings. After publication of a bulletin on clipper.com, The Clipper Group offers all vendors and users the opportunity to license its publications for a fee, since linking to Clipper's web pages, posting of Clipper documents on other's websites, and printing of hard-copy reprints is not allowed without payment of related fee(s). Less than half of our publications are licensed in this way. In addition, analysts regularly receive briefings from many vendors. Occasionally, Clipper analysts' travel and/or lodging expenses and/or conference fees have been subsidized by a vendor, in order to participate in briefings. The Clipper Group does not charge any professional fees to participate in these information-gathering events. In addition, some vendors sometime provide binders, USB drives containing presentations, and other conference-related paraphernalia to Clipper's analysts.

Regarding the Information in this Issue

The Clipper Group believes the information included in this report to be accurate. Data has been received from a variety of sources, which we believe to be reliable, including manufacturers, distributors, or users of the products discussed herein. The Clipper Group, Inc., cannot be held responsible for any consequential damages resulting from the application of information or opinions contained in this report.