

Don't Let Automation Water Down Your Data Center



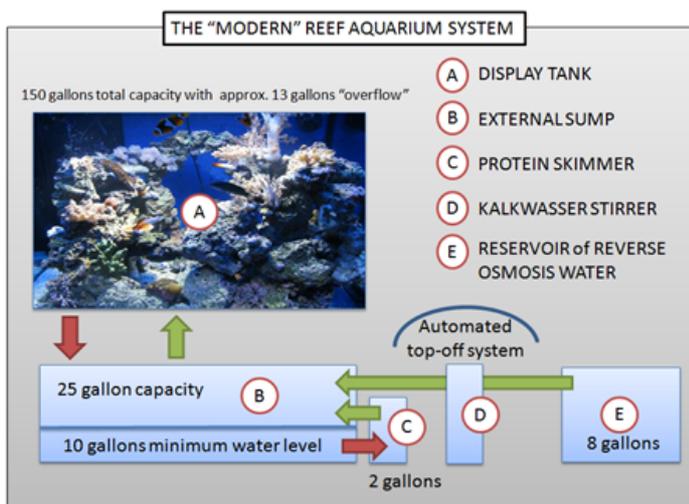
Lori MacVittie, 2011-11-05

A recent power outage in the middle of the night reveals automation without context can be expensive for aquariums – and data centers.

You may recall from several posts ([Cloud Chemistry 101](#), [The Zero-Product Property of IT](#) and [The Number of the Counting Shall be Three \(Rules of Thumb for Application Availability\)](#)) that one of my hobbies is “reefing.”

No, it's not that kind of reefer madness, it's the *other* kind – the kind associated with aquariums and corals and all manner of strange looking ocean-living fish. I only recently re-engaged after years of avoiding the hassle (and enjoyment) and have been learning a lot, especially in terms of what's been learned by others during my off-years.

When I set up my most recent aquarium adventure – a 150 gallon reef – I decided to use an external sump. Think of it like an external reservoir in which all sorts of interesting filtering and water quality activities can be handled without all the tubes and equipment that might otherwise clutter up the main display tank. It's also handy for setting up things like automated top-off systems, which automatically add fresh water to the system to compensate for evaporation. Such



systems can, it turns out, be problematic if they aren't enabled with the proper context in which to automatically kick in. It's loose coupling of a system, much in the same way application delivery can abstract policy enforcement and infrastructure services from the applications it delivers, making the system more agile and able to be adapted to problems without disrupting the main tank, er, application.

MAINTAINING EQUILIBRIUM

I talked about achieving dynamic equilibrium in a previous post so

suffice to say that as water evaporates from an aquarium *only water is lost*. This is not so critical a point in a freshwater system, but in a salt-water system it is absolutely important to understand.

The more “salt per gallon” in a salt-water aquarium, the higher the salinity. Obviously if salt is not evaporating but water is, then salinity increases. Conversely, if too much fresh water (i.e. 0 salinity) is added, salinity decreases. You might guess that a rather narrow range of salinity is required to support a reef. Too low or too high, and things start suffering rather quickly. The balance needs to be maintained in order to maintain a healthy ecosystem.

Now, the relationship between all the moving parts in my reef setup are very much like the complex relationships between components and resources in a data center. Water (requests and responses) flow out of the display tank and into the sump (application delivery controller), are filtered by a protein skimmer (web application firewall), and then returned via a return line to the main tank. As water evaporates it reaches a minimum level on an automated sensor (application health monitors) that trigger a response that forces additional fresh water (compute resources) into the sump, which re-establishes equilibrium and maintains salinity by sustaining a specific water-salt ratio in the ecosystem. Flow rates are equalized between input and output, and when the power is on everything runs smooth as pie.

But when power went out not once, but three times last week, that automation that saves me so much time under normal operations, bit me in the proverbial derriere.

AN EXCESS of RESOURCES

So what happens when the power goes out? Well, if not for the protein skimmer (C in the lovely diagram to the left) nothing. The pumps stop pumping and water in the display tank (A) which continues to drain from the overflow until it hits the siphon break and then stops.

That's about 13 gallons of water. When added to the sump's level of 10 gallons, that's about 23 gallons of a 25 gallon capacity container. But add in the approximately 2 gallons from the protein skimmer combined with natural water displacement from the equipment and ... wet floor. Water overflow. Once might not be too bad, but twice? Three times in one night?

But it wasn't just the wet floor that was the problem. See, once the power returned the automated top-off system, recognizing the water level was down, did what it does best: pumped fresh, desalinated water into the sump. It did it so well, in fact, that the salinity levels in the entire system dropped from a comfortable 1.025 to a rock-bottom minimum of 1.023. Luckily that's not "rock-bottom" in terms of survival, and everything that was alive is still doing well and in fact flourishing, but it pointed out a flaw with the automation I'd put into place – it's not contextually aware. It's not intelligent. It just ... is.

A more experienced reefer (with these kinds of complex systems) would point out that salinity monitoring is essential and that a secondary system designed to ensure the maintenance of a specific specific gravity (another way to say salinity) is vital to maintaining the proper water chemistry. I would be inclined to agree after recent events, and find that this is a fine example of potentially similar problems with data center automation.

CONTEXT is CRITICAL

Let's assume a data center that uses monitoring of application performance and has in place an auto-scaling or server flexing system that, when triggered, automatically adds new resources to a given application to improve performance, assuming resource consumption (high volume of users) is the core factor in performance.

That all sounds great in theory, like my reef setup, but in practice it can go horribly wrong. For example, if the reason resource availability is decreasing is due to a concerted [DoS attack across multiple layers of the stack](#), adding *more* resources is unlikely to restore equilibrium. You can add compute resources all day but it won't address the consumption of *bandwidth* or infrastructure resources caused by the attack. Without context, the automated system simply does without thinking what it's been told to do. And if those resources are in a cloud-based environment for which you are charged by the instance hour, you may increase costs dramatically without seeing any return on that investment. Cloud-bursting can be a valuable tactical response to balancing the need for more capacity with costs, but if those resources are added without context then, like adding fresh-water to compensate for non-evaporative water loss to a salt-water system, you may be diluting the efficiency of the entire application delivery chain.

But if the automated system had visibility and context-awareness, if it was intelligent and could factor in all the variables – network and compute – it could react accordingly and perhaps take some *other* action that would address the real problem, like activating security-minded policies that throttle bandwidth based on usage patterns, or start blocking offending user sessions. The what is less important than how, for our purposes, because it's really about having the context in the first place to enable the application of organizational-specific policies supporting operational goals. Without context, without collaboration, automation is likely to result in blind decisions – made without understanding the root cause and potentially causing more damage than good.

Context is critical to ensure that automation is supportive of – not detrimental to- operational efficiency and goals.

- [Cloud Chemistry 101](#)
- [The Zero-Product Property of IT](#)
- [The Number of the Counting Shall be Three \(Rules of Thumb for Application Availability\)](#)
- [Operational Risk Comprises More Than Just Security](#)
- [Aligning IT with the Business by Decreasing Efficiency](#)
- [Cloud Needs Context-Aware Provisioning](#)

-  [Service Virtualization Helps Localize Impact of Elastic Scalability](#)
-  [What is a Strategic Point of Control Anyway?](#)
-  [Are You Ready for the New Network?](#)
-  [All Infrastructure 2.0 Topics on DevCentral](#)
-  [I Find Your Lack of Win Disturbing](#)

F5 Networks, Inc. | 401 Elliot Avenue West, Seattle, WA 98119 | 888-882-4447 | f5.com

F5 Networks, Inc.
Corporate Headquarters
info@f5.com

F5 Networks
Asia-Pacific
apacinfo@f5.com

F5 Networks Ltd.
Europe/Middle-East/Africa
emeainfo@f5.com

F5 Networks
Japan K.K.
f5j-info@f5.com