# Modular Systems: The Evolution of Reliability

## White Paper 76

Revision 1

by Neil Rasmussen and Suzanne Niles


## > Executive summary

Nature proved early on that in complex systems, modular designs are the ones that survive and thrive. An important contributor to this success is the critical reliability advantage of *fault tolerance*, in which a modular system can shift operation from failed modules to healthy ones while repairs are made. In data centers, modular design has already taken root in new fault-tolerant architectures for servers and storage systems. As data centers continue to evolve and borrow from nature's blueprints, data center physical infrastructure (DCPI) must also evolve to support new strategies for survival, recovery, and growth.


## Contents

*Click on a section to jump to it*


| | |
|---|---|
| Introduction | 2 |
| Nature's case study: Early life | 2 |
| IT case study: Disk drives | 3 |
| IT case study: Blade servers | 4 |
| The changing definition of failure for IT systems | 6 |
| Implications for DCPI | 7 |
| Conclusion | 9 |
| Resources | 10 |


APC white papers are now part of the Schneider Electric white paper library produced by Schneider Electric's **Data Center Science Center**
DCSC@Schneider-Electric.com

# Introduction

Modularity is an established technique for organizing and simplifying a complex system. From elementary (flashlight batteries) to complex (the cells of an organism), modularity has a record of success that is hard to challenge. Nonetheless, in man-made systems on the brink of the evolutionary transition from monolithic to modular design, there can be skepticism and slow starts until modularity settles in and begins to deliver its time-tested benefits.

Data center physical infrastructure (DCPI) of data centers is in this transition phase. While the physically evident attributes of building-block architecture – scalability, flexibility, simplicity, portability – are easily understood and not in serious dispute, one aspect of modular design in this industry has become a subject for debate: reliability.

Applying classic, simple reliability analysis to this new way of doing things ("more parts equals greater risk of failure") is at best incomplete, at worst misleading. The mission of this paper is to illustrate, through case studies, how modularity not only delivers its more obvious and easily understood benefits but also its most subtle, least understood, and profound reliability benefit: *fault tolerance*. The inherent fault tolerance of modular design provides a powerful new defense against failure, introducing into complex systems a strategy for reliability that is not only adequate, but superior.

# Nature's case study: Early life

The history of modularity is much older than data centers or flashlight batteries. Very early non-modular systems – single-celled organisms – lived on Earth three billion years ago. The fossil record of these organisms reveals that they developed shells, tentacles, mouths, arms, grippers, and a host of other intricate structures. Some grew to surprising sizes, up to six inches (15 centimeters) across. These complex monolithic single-celled designs dominated Earth's elemental food chain for billions of years.

Then, some 500 million years ago, multi-cellular organisms came into existence. In mere tens of millions of years, they evolved so rapidly that they overtook three billion years of evolution of the complicated single-celled organisms, replacing them as the dominant design.

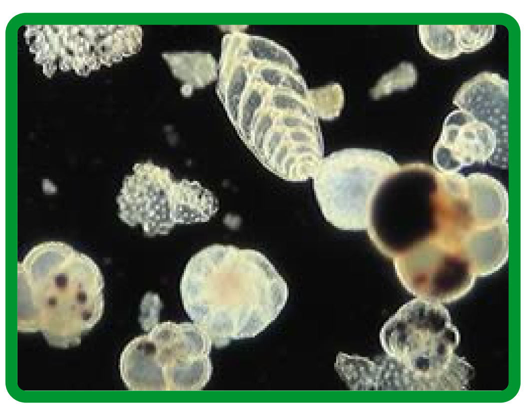

**Figure 1**

*Early complex single-celled life*

## The modular advantage for multi-celled organisms

Why did the modular, multi-celled design prevail over the entrenched monolithic design?

- **Ability to scale and grow.** System growth, both in size and in addition of new capabilities, was accomplished simply by adding new modules (cells) that could interact with existing ones using standard interfaces.

- **Simpler process of duplication.** Duplicating a number of smaller, less complicated cells was easier, faster, and more reliable than duplicating a single complicated one.
- **Ability to specialize the function of modules.** Delegation and specialization of cell tasks provided the same effectiveness and efficiencies inherent in teamwork. In the early multi-celled organisms, one kind of cell could be for locomotion, another kind for protection, another kind for sensing food, and so on.
- **Rapid adaptation to the environment.** By adding, subtracting, or modifying cells, incremental design changes could be more quickly tried and either adopted or rejected.
- **Fault tolerance.** With cell redundancy, individual cells could fail without degrading the system, allowing for concurrent cell repair without system downtime (disability or death in this case).

**Figure 2**

*Early multi-celled life*

The last attribute above, fault tolerance, is a critical reliability advantage of modular systems over monolithic systems. Modularity "packages" a system into smaller pieces, which facilitates redundancy of component parts so that failure of one, or even several, need not adversely affect operation of the system. With a simple scratch, human skin can lose hundreds of cells, yet our bodies don't fail from such a loss. Other cells carry on while repairs are made. We humans didn't invent modularity – we *are* modularity. With trillions of modules (cells) per person, we personally enjoy the benefits of fault tolerance every day.

## IT case study: Disk drives

In the mainframe days of data centers, storage devices were large proprietary hard disks, with stacks of 14-inch metal platters, elaborate read/write mechanisms, and enclosures the size of washing machines. In 1978, IBM patented the idea of using arrays of smaller disks, but didn't go forward with it because they felt it could never be as reliable as the conventional monolithic designs. The study and practice of fault tolerance was in its infancy, primarily confined to the aerospace industry where component failure in electronic systems could cost lives.[1]

In 1987, Berkeley researchers noted the widening gap between computing speed and storage access speed, and saw the emergence of external disk drives for personal computers as an opportunity to use them as building blocks for a system with faster data transfer. A year later they presented a landmark paper, "A Case for Redundant Arrays of Inexpensive Disks (RAID)," proposing several data-writing schemes ("RAID levels") that such arrays could use

[1] Today, with IT operations at the heart of nearly every industry, including health and the military, data centers can be mission-critical to the point that failure has the potential for loss of life. Fault tolerance is therefore becoming relevant to their design even beyond what is desired by economic interests.

to store, retrieve, *and recover* data. In 1990, theory and hardware came together using the personal computer industry's 5.25-inch disks, which had evolved to the point where they had the capacity, performance, and reliability to be used in the first RAID arrays. These new modular storage devices offered a choice of tradeoffs between redundancy and read/write speed, and occupied a fraction of the floor space of the mainframe storage devices they replaced.

**Figure 3**

*RAID array*

### The modular advantage for RAID arrays

Why have modular RAID arrays prevailed over the old monolithic storage devices?

- **Ability to scale and grow.** Storage capacity can be easily increased by increasing the number of modules per array, or by adding arrays.
- **Simpler process of duplication.** It is much easier to manufacture the many small drives that serve as RAID modules than it is to manufacture the old complicated large drives.
- **Ability to specialize the function of modules.** The individual drives of an array can be used for additional storage capacity, increased access speed, or greater redundancy, depending upon the RAID level defined for the array. In addition, the RAID arrays themselves can be considered modules at a higher level, with a different application assigned to each RAID array.
- **Rapid adaptation to the environment.** Drives can be added or removed, and the RAID level can be easily changed for the desired tradeoffs in capacity, speed, and redundancy.
- **Fault tolerance.** RAID data-writing schemes incorporate redundancy that provides the ability to recover data when one of the drives fails.

In a surprise to its designers, RAID's enthusiastic reception in the marketplace was driven not so much by its increased speed – the original goal of the design – but by the increased reliability that resulted from fault tolerance. Until the authors of the 1988 paper showed the fault tolerance possibilities of the RAID design – during live presentations they would simply remove one drive while the array continued to function – the prevailing notion had been the typical, but erroneous, pre-fault-tolerance understanding of reliability: a multiple-drive system had to be less reliable because it had more parts.

## IT case study: Blade servers

Blade servers are at the center of a transition to modular design that is in process as this paper is being written. For many years, traditional standalone servers grew larger and faster, taking on more and more tasks as networked computing expanded. New servers were added to data centers as the need arose, often as a quick fix with little coordination or planning; it was not unusual for data center operators to discover that servers had been added without their knowledge. The resulting complexity of boxes and cabling became a growing invitation to confusion, mistakes, and inflexibility.

Blade servers, first appearing in 2001, are a very simple and pure example of modular architecture – the blades in a blade server chassis are physically identical, with identical processors, ready to be configured and used for any purpose desired by the user. Their introduction brought many benefits of modularity to the server landscape – scalability, ease of duplication, specialization of function, and adaptability.

But while these classic modular advantages have given blade servers a growing presence in data centers, their full potential awaits the widespread implementation of one remaining critical capability of modular design: fault tolerance. Fault tolerant blade servers – ones with built-in "failover" logic to transfer operation from failed to healthy blades – have only recently started to become available and affordable. The reliability of such fault tolerant servers will surpass that of current techniques involving redundant software and clusters of single servers, putting blade servers in a position to become the dominant server architecture of data centers. With the emergence of automated fault tolerance, industry observers predict rapid migration to blade servers over the next five years.

**Figure 4**

*Conventional servers*

## The modular advantage for blade servers

Why will modular blade servers prevail over larger, standalone servers?

- **Ability to scale and grow.** Computing capacity can be easily increased by adding more modules (blades).
- **Simpler process of duplication.** It is much easier to manufacture many small blades than entire servers. Power supply, cooling fans, network connections, and other support components are centralized in the chassis and shared by the blades, so blade structure is simplified.
- **Ability to specialize the function of modules.** Individual blades can be configured with software applications as desired by the user.
- **Rapid adaptation to the environment.** Blades can be added or removed as required by business or financial requirements, and blades can be reconfigured to run different applications.
- **Fault tolerance.** Failure of a blade can be handled automatically by built-in "failover" logic that seamlessly transfers operation to other blades.

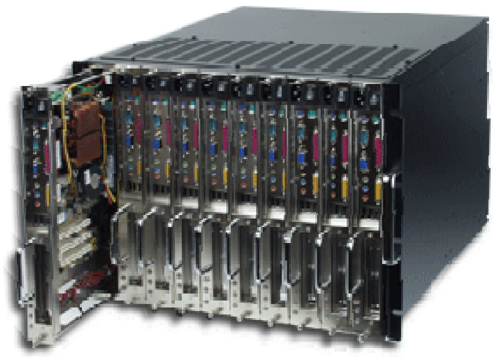

**Figure 5**

*Blade server (10 blades in chassis)*

## The changing definition of failure for IT systems

These case studies show how modular design prevails over complex monolithic design, for fundamental reasons that are intrinsic to the nature of modularity. One of these reasons, fault tolerance, has profound significance for the future of data centers. Once servers and storage devices become fault tolerant throughout the data center, it will change the way IT failure is defined.

Consider two different failure scenarios in a data center (**Figure 6**). On the right is a failure of all the racks, as would happen if a single large UPS protecting the entire data center failed and dropped the load. On the left is a failure of one rack. In conventional data centers, these two scenarios would be viewed by IT managers as the same failure, because – in a one-rack failure – interdependencies among servers, disk arrays, switches, and routers would likely cause cascading effects that bring down the entire data center.

As the new array-style modular designs for computing and storage take hold, the failure on the left – one rack – is beginning to be viewed by IT managers as a "better" failure, because redundancy of resources now offers the possibility of data center survival even when individual units fail. As fault tolerant architectures become more widespread, data centers will be able to tolerate a greater number of unit failures without total system failure. When blade servers fulfill their early promise of seamless fault tolerance, the failure of one, two, three, or even more racks will be a survivable event.

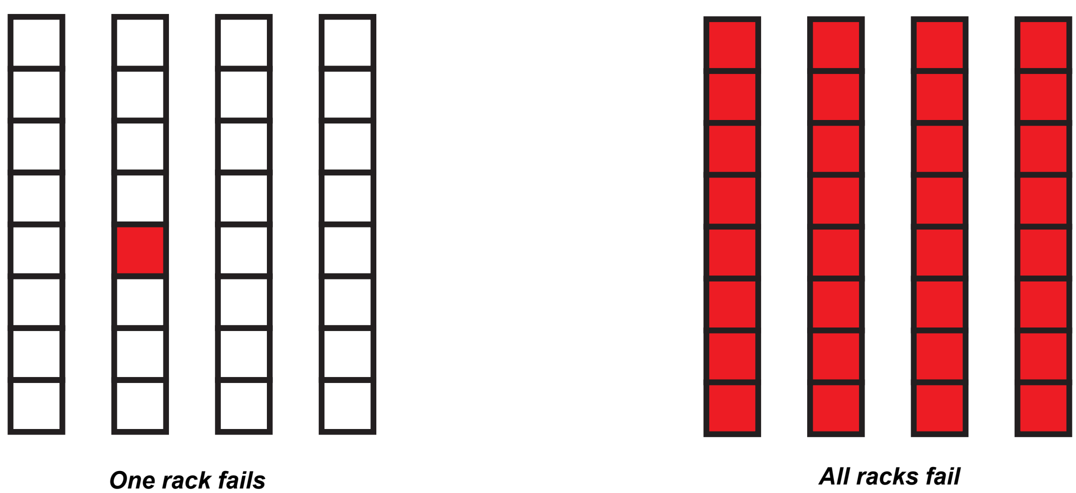


**Figure 6**

*Two failure scenarios for a data center (overhead view, four rows of eight racks)*

## Implications for DCPI

This new paradigm for failure management – the expectation that some modules will inevitably fail, combined with robust preparedness for surviving it – has implications for how the new IT architecture should be protected by its data center physical infrastructure. For example, as data centers become more fault tolerant in their IT layer, power protection by a single, large UPS will become less optimal since failure of that UPS brings down the whole system – an unnecessary outcome in a fault-tolerant data center capable of surviving the loss of a rack. If UPSs are distributed throughout the data center, one UPS for every rack, then failure of any single UPS will fail only one rack, not the whole system. Even though there are more UPSs, which increases the likelihood of individual UPS failure, such a failure can be tolerated by the system. If it takes failure of three racks to fail the entire system, then three of those UPSs would have to fail simultaneously to bring down the system, an extremely unlikely event – much less likely than failure of a single large UPS. For this reason, reliability theory strongly favors modular distributed power and cooling architecture as IT systems become more fault tolerant.

### Monolithic vs. modular DCPI

The architecture of data center physical infrastructure (DCPI) has remained largely unchanged over the 30-year history of data centers. From the smallest computer rooms to the largest enterprise facilities, the persistent model for physical infrastructure has been a centralized "plant" for power protection and cooling. The engineering of this kind of infrastructure results in a monolithic, unique configuration of equipment and connections. By replacing such architecture with modular design, not only can DCPI properly support modular, fault-tolerant IT equipment, but the DCPI equipment itself can enjoy the advantages of modularity – including the reliability advantages of fault tolerance.

**Figure 7**

*Centralized monolithic UPS*

### The modular advantage for DCPI

Why will modular DCPI replace conventional monolithic DCPI?

- **Ability to scale and grow.** Modular DCPI can be sized to align with the data center's present IT requirements, and grow as requirements dictate. This advantage has particular importance to DCPI, where the traditional method has been one-time deployment of power and cooling to support projected maximum IT requirements, which results in significant waste in both capital and operating expenditures.

- **Simpler process of duplication.** Modular design means manufacturing a large number of small units, instead of a small number of large units. Greater production volume means fewer defects; smaller, simpler design means more automation and less manual work during manufacture, which means fewer defects.
- **Ability to specialize the function of modules.** Power-protection and cooling units can be manufactured in a variety of configurations to target the particular availability and cooling requirements of different parts of the data center.
- **Rapid adaptation to the environment.** With new equipment being added and IT equipment changing every 2 to 3 years, the contents of data centers are under constant revision. New equipment might have different sizes or shapes, different power or cooling requirements, different plugs, and so on. Modular DCPI can easily be scaled up or reconfigured to meet these changing IT needs.
- **Fault tolerance.** Just as fault tolerant IT equipment allows continued data center operation when an IT component fails, fault tolerant DCPI equipment allows continued operation of power or cooling when a DCPI component fails. Fault tolerance can be accomplished by redundancy of DCPI units, or by internal redundancy of components *within* DCPI units – for example, by having extra power modules in a UPS.

Just as in the previous case studies of modular design, the first four attributes above are instrumental in the success of the design, but the fifth – fault tolerance – is critical. Further, since the data center depends absolutely on power and cooling for its operation, fault-tolerant reliability is as critical in DCPI as it is in the IT equipment it protects. A fault-tolerant data center without fault-tolerant DCPI will make no more sense than a suspension bridge with a strong roadbed but weak cables.

**Figure 8**

*Rack-level modular UPS*

## Conclusion

The design transition from monolithic to modular is a natural evolution for complex systems because of the advantages it provides in efficiency, flexibility, and reliability. Examining success stories makes it easier to see the potential of modularity to make significant, even revolutionary, improvements to systems that have been monolithic since their inception and have never been understood any other way. Fault tolerance and other critical attributes of modularity – the ability to scale, adapt, specialize, and duplicate – are as evident, and inevitable, in man-made modular systems as they have been in natural ones.

The IT world has already seen these advantages in the emergence of modular designs for storage and computing – RAID arrays and blade servers. Even more significant, data centers are now poised to follow industries such as aerospace in system-wide deployment of a modularity advantage that has routinely been used in mission-critical systems since the 1970s: fault tolerance. Fault tolerance recognizes that careful control of component quality is only the first step toward system reliability, and that continued system operation in the face of component failure is the ultimate reliability tactic.

As modularity and fault tolerance become the new models for data center design, data center physical infrastructure must also move in the same direction, both to protect these data centers effectively and to gain modularity's benefits for its own efficiency, flexibility, and reliability.


## About the author

**Neil Rasmussen** is a Senior VP of Innovation for Schneider Electric. He establishes the technology direction for the world's largest R&D budget devoted to power, cooling, and rack infrastructure for critical networks.

Neil holds 19 patents related to high-efficiency and high-density data center power and cooling infrastructure, and has published over 50 white papers related to power and cooling systems, many published in more than 10 languages, most recently with a focus on the improvement of energy efficiency. He is an internationally recognized keynote speaker on the subject of high-efficiency data centers. Neil is currently working to advance the science of high-efficiency, high-density, scalable data center infrastructure solutions and is a principal architect of the APC InfraStruXure system.

Prior to founding APC in 1981, Neil received his bachelors and masters degrees from MIT in electrical engineering, where he did his thesis on the analysis of a 200MW power supply for a tokamak fusion reactor. From 1979 to 1981 he worked at MIT Lincoln Laboratories on flywheel energy storage systems and solar electric power systems.

# Resources

*Click on icon to link to resource*

**Data Center Physical Infrastructure: Optimizing Business Value**
White Paper 117

**Standardization and Modularity in Data Center Physical Infrastructure**
White Paper 116

**White Paper Library**
whitepapers.apc.com

**TradeOff Tools™**
tools.apc.com

## Contact us

For feedback and comments about the content of this white paper:

Data Center Science Center
**DCSC@Schneider-Electric.com**

If you are a customer and have questions specific to your data center project:

Contact your **Schneider Electric** representative