



Data Center Design Philosophy

By Rob Snevely - Enterprise Technology Center

Sun BluePrints™ OnLine - January 2002



<http://www.sun.com/blueprints>

Sun Microsystems, Inc.
901 San Antonio Road
Palo Alto, CA 94303 USA
650 960-1300 fax 650 969-9131

Part No.: 816-2765-10
Revision 1.0, 01/08/02
Edition: January 2002

Copyright 2002 Sun Microsystems, Inc. 901 San Antonio Road, Palo Alto, California 94303 U.S.A. All rights reserved.

This product or document is protected by copyright and distributed under licenses restricting its use, copying, distribution, and decompilation. No part of this product or document may be reproduced in any form by any means without prior written authorization of Sun and its licensors, if any. Third-party software, including font technology, is copyrighted and licensed from Sun suppliers.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, Sun BluePrints, and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries.

The OPEN LOOK and Sun™ Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

RESTRICTED RIGHTS: Use, duplication, or disclosure by the U.S. Government is subject to restrictions of FAR 52.227-14(g)(2)(6/87) and FAR 52.227-19(6/87), or DFAR 252.227-7015(b)(6/95) and DFAR 227.7202-3(a).

DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

Copyright 2002 Sun Microsystems, Inc., 901 San Antonio Road, Palo Alto, Californie 94303 Etats-Unis. Tous droits réservés.

Ce produit ou document est protégé par un copyright et distribué avec des licences qui en restreignent l'utilisation, la copie, la distribution, et la décompilation. Aucune partie de ce produit ou document ne peut être reproduite sous aucune forme, par quelque moyen que ce soit, sans l'autorisation préalable et écrite de Sun et de ses bailleurs de licence, s'il y en a. Le logiciel détenu par des tiers, et qui comprend la technologie relative aux polices de caractères, est protégé par un copyright et licencié par des fournisseurs de Sun.

Des parties de ce produit pourront être dérivées des systèmes Berkeley BSD licenciés par l'Université de Californie. UNIX est une marque déposée aux Etats-Unis et dans d'autres pays et licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, Sun BluePrints, et Solaris sont des marques de fabrique ou des marques déposées, ou marques de service, de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays.

L'interface d'utilisation graphique OPEN LOOK et Sun™ a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciés de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui en outre se conforment aux licences écrites de Sun.

CETTE PUBLICATION EST FOURNIE "EN L'ETAT" ET AUCUNE GARANTIE, EXPRESSE OU IMPLICITE, N'EST ACCORDEE, Y COMPRIS DES GARANTIES CONCERNANT LA VALEUR MARCHANDE, L'APTITUDE DE LA PUBLICATION A REpondre A UNE UTILISATION PARTICULIERE, OU LE FAIT QU'ELLE NE SOIT PAS CONTREFAISANTE DE PRODUIT DE TIERS. CE DENI DE GARANTIE NE S'APPLIQUERAIT PAS, DANS LA MESURE OU IL SERAIT TENU JURIDIQUEMENT NUL ET NON AVENU.



Please
Recycle



Adobe PostScript

Data Center Design Philosophy

Editor's Note – This article is the complete first chapter of the Sun BluePrints™ book, "Enterprise Data Center Design and Methodology", by Rob Snevely (ISBN 0-13-047393-6), which is available through www.sun.com/books, amazon.com, fatbrain.com and Barnes & Noble bookstores.

"Form follows function."

- Louis Henri Sullivan

The detailed process of data center design appears on the outset to be a purely mechanical process involving the layout of the area, computations to determine equipment capacities, and innumerable other engineering details. They are, of course, essential to the design and creation of a data center, however, the mechanics alone do not a data center make. The use of pure mechanics rarely creates anything that is useful, except perhaps by chance.

There are, in fact, some philosophical guidelines that should be kept in mind during the data center design process. These are based on the relatively short history of designing and building practical data centers, but are also based on design concepts going way back. This chapter looks at some of these philosophies.

This chapter contains the following sections:

- Look Forward by Looking Back
- A Modern Pantheon
- Fundamentals of the Philosophy
- Top Ten Data Center Design Guidelines

Look Forward by Looking Back

The idea that technology is relatively new, that it arose within the last fifty to one hundred years, is a common misconception. There have been great advances, particularly in the electronic age, but the truth of the matter is that technology has been around since human beings began bashing rock against rock.

One of the most interesting things about design is that it draws from many sources. Paintings by Raphael and Botticelli in the Renaissance were dependent on the mathematics of perspective geometry developed more than a millennia and a half before either were born. They also drew on the language and form of classical architecture and Greco-Roman mythology to provide settings for many of their works. Raphael and Botticelli created works that had never been seen before, but they could not have done this without the groundwork that had been set down in the previous centuries.

Look back to the most prolific designers and engineers in the history of western civilization: The Romans. Roman advances in design and technology are still with us today. If you cross a bridge to get to work, or take the subway, or walk down the street to get a latte, chances are you are doing so using elements of Roman design and technology. These elements are the arch and concrete.

When entering the Pantheon in Rome, most people probably don't remark, "What a great use of the arch!" and "That dome is a single concrete structure." However, without the modular design of the arch and the invention of concrete, the Roman Pantheon could not have been built.

The Romans understood that the arch, by design, had strength and the ability to transfer load from its center down to its base. They had used the arch in modular and linear ways to build bridges and carry water for their water systems. But in the Pantheon, the modularity of the arch realized its true potential. Spin an arch at its center point and you create a dome. This means that across any point in the span you have the strength of the arch. Also, they had found that concrete could be used to bond all of these arches together as a single dome. Concrete allowed this dome structure to scale beyond any other dome of its time. It would take eighteen centuries for technology to advance to the point where a larger dome than that of the Pantheon could be built.

What does the architecture of ancient Rome have to do with data centers? The physical architecture itself has little in common with data centers, but the design philosophy of this architecture does. In both cases, new ideas on how to construct things were needed. In both cases, using the existing design philosophies of the time, "post and lintel" for ancient Rome, and "watts per square foot" for data

centers, would not scale to new requirements. It is this idea, the design philosophy of modular, scalable units, that is critical to meet the requirements of today's data centers and, more importantly, the data centers of the future.

A Modern Pantheon

A modern data center still shares many aspects with ancient architecture, structurally and in service. The form literally follows the function. The purpose of both the Pantheon and a data center is to provide services. To provide services, its requirements for continual functioning must be met. This is the design team's primary concern. The design of the data center must revolve around the care and feeding of the service providing equipment.

These functional requirements of the data center are:

- A place to locate computer, storage, and networking devices safely and securely
- To provide the power needed to maintain these devices
- To provide a temperature-controlled environment within the parameters needed to run these devices
- To provide connectivity to other devices both inside and outside the data center

In the design philosophy of this book, these needs must be met and in the most efficient way possible. The efficiency of the data center system relies entirely on the efficiency of the design. The fundamental principles of a data center philosophy should be your guiding principles.

The phrase "design philosophy" could have many different meanings. For the purposes of this book we'll use the following definition: A design philosophy is the application of structure to the functional requirements of an object based on a reasoned set of values.

Fundamentals of the Philosophy

There are five core values that are the foundation of a data center design philosophy: simplicity, flexibility, scalability, modularity, and sanity. The last one might give you pause, but if you've had previous experience in designing data centers, it makes perfect sense.

Design decisions should always be made with consideration to these values.

Keep the Design as Simple as Possible

A simple data center design is easier to understand and manage. A basic design makes it simple to do the best work and more difficult to do sloppy work. For example, if you label everything—network ports, power outlets, cables, circuit breakers, their location on the floor—there is no guess work involved. When people set up a machine, they gain the advantage of knowing ahead of time where the machine goes and where everything on that machine should be plugged in. It is also simpler to verify that the work was done correctly. Since the locations of all of the connections to the machine are pre-labeled and documented, it is simple to record the information for later use, should the machine develop a problem.



FIGURE 0-1 Simple, Clean, Modular Data Center Equipment Room

Design for Flexibility

Nobody knows where technology will be in five years, but it is a good guess that there will be some major changes. Making sure that the design is flexible and easily upgradable is critical to a successful long-term design.

Part of flexibility is making the design cost-effective. Every design decision has an impact on the budget. Designing a cost effective data center is greatly dependent on the mission of the center. One company might be planning a data center for mission critical applications, another for testing large-scale configurations that will go into a mission critical data center. For the first company, full backup generators to drive the

entire electrical load of the data center might be a cost-effective solution. For the second company, a UPS with a 20-minute battery life might be sufficient. Why the difference? If the data center in the first case goes down, it could cost the company two million dollars a minute. Spending five million on full backup generators would be worth the expense to offset the cost of downtime. In the second case, the cost of down time might be \$10,000 an hour. It would take 500 hours of unplanned downtime to recoup the initial cost of five million dollars of backup generators.

Design for Scalability

The design should work equally well for a 2,000, 20,000, or 2,000,000 square foot data center. Where a variety of equipment is concerned, the use of watts per square foot to design a data center does not scale because the needs of individual machines are not taken into consideration. This book describes the use of rack location units (RLUs) to design for equipment needs. This system is scalable and can be reverse-engineered.

Use a Modular Design

Data centers are highly complex things, and complex things can quickly become unmanageable. Modular design allows you to create highly complex systems from smaller, more manageable building blocks.

These smaller units are more easily defined and can be more easily replicated. They can also be defined by even smaller units, and you can take this to whatever level of granularity necessary to manage the design process. The use of this type of hierarchy has been present in design since antiquity.

Keep Your Sanity

Designing and building a data center can be very stressful. There are many things that can, and will, go wrong. Keep your sense of humor. Find ways to enjoy what you're doing. Using the other four values to evaluate design decisions should make the process easier as they give form, order, and ways to measure the value and sense of the design decisions you're making. Primarily, they help to eliminate as many unknowns as possible, and eliminating the unknowns will make the process much less stressful.

Top Ten Data Center Design Guidelines

The following are the top ten guidelines selected from a great many other guidelines, many of which are described throughout this book.

1. **Plan ahead.** You never want to hear “Oops!” in your data center.
2. **Keep it simple.** Simple designs are easier to support, administer, and use. Set things up so that when a problem occurs, you can fix it quickly.
3. **Be flexible.** Technology changes. Upgrades happen.
4. **Think modular.** Look for modularity as you design. This will help keep things simple and flexible.
5. **Use RLUs, not square feet.** Move away from the concept of using square footage of area to determine capacity. Use RLUs to define capacity and make the data center scalable.
6. **Worry about weight.** Servers and storage equipment for data centers are getting denser and heavier every day. Make sure the load rating for all supporting structures, particularly for raised floors and ramps, is adequate for current and future loads.
7. **Use aluminum tiles in the raised floor system.** Cast aluminum tiles are strong and will handle increasing weight load requirements better than tiles made of other materials. Even the perforated and grated aluminum tiles maintain their strength and allow the passage of cold air to the machines.
8. **Label everything.** Particularly cabling! It is easy to let this one slip when it seems as if “there are better things to do.” The time lost in labeling is time gained when you don’t have to pull up the raised floor system to trace the end of a single cable. And you *will* have to trace bad cables!
9. **Keep things covered, or bundled, and out of sight.** If it can’t be seen, it can’t be messed with.
10. **Hope for the best, plan for the worst.** That way, you’re never surprised.

Author's Bio: Rob Snevely

Rob Snevely is an Enterprise Architect at Sun Microsystems, Inc. working in the Quality Engineering and Deployment organization. He has over 14 years experience working with large scale UNIX® systems in data center environments, and is responsible for data center architecture for all of Sun's Enterprise Technology Centers. Since coming to work for Sun in 1990 as a system administrator, he has been involved with network and system performance and large scale system engineering. His liberal arts background in theatre, art history, and fashion design augment his practical and pragmatic methods for designing data centers.