

Meet CFUS



Joseph C. (Joe) Hibbitt
Principal, President
Los Angeles, California



Manny Mangahas
Principal, VP—East Coast Operations
Clifton, Virginia (Washington DC)



Burnie Reed
Principal, VP—Midwest Operations
Dallas, Texas

CFUS Update!

Industry: Defense

Developed and implemented processes and procedures for capacity planning, performance tuning, user security, disaster recovery, and upgrades on UNIX and Oracle systems.

**Service: System Administration
Technology : Oracle**

Industry: Defense

Provided Oracle developer report writers, administrators, and project managers to support a custom Oracle financial system conversion

Service: IT Staff Augmentation

The Pros and Cons of Web Applications

There has been a long running debate about web applications replacing desktop software applications. While some functions are better suited to web applications. It is my belief that security concerns and legacy systems will prevent desktop software from becoming obsolete.

Some argue that the debate between web applications and desktop applications is pointless as there is no clear answer. While still others argue that the issue at hand is as much a business and marketing issue, as it is a technological issue.

What Defines a Web Application Vs a Desktop Application?

A web application is an application delivered to users from a web server like the Internet. Some businesses run web applications on an intranet, as well. Web applications are becoming more popular due to the widespread use of the web browser as a client.

Some applications are better suited and more likely to become successful as web applications. Web applications designed specifically for search

engine optimization, have become increasingly popular. It is easy to understand why web applications that relate to the Internet would prosper, while business applications may have less appeal in a web environment.

A desktop application is a self-contained program that performs a defined set of tasks under the user control. Desktop applications run from a local drive and do not require a network or connectivity to operate or function properly, though if attached to a network desktop applications might use the resources of the network.

Pros and Cons to Desktop and Web Applications:

Easily Accessible

Web applications can be easily accessed from any computer or location that has Internet access. Travelers especially benefit from the accessibility. This often means that if a traveler has access to a computer, phone or handheld with Internet connectivity they can utilize the web application.

Low Maintenance & Forced Upgrades

Desktop applications need to be individually installed on each computer, while web applications require a single installation.

Many web applications are hosted by a 3rd party and the maintenance fall under the applications hosts responsibility. The ability to update and maintain web applications without distributing and installing software on potentially thousands of client computers is a key reason for the popularity of web based applications. This can be a blessing and a curse as users of web applications on hosted systems are at the mercy of the host, if an upgrade does not go well, or the individual user doesn't want or need the new features the upgrade will still go forward.

Increased Security Risks

There are always risks involved when dealing with working online, regardless of how secure a host might say a web application is, that fact of the matter stands that the security risk of running an application of the Internet is more significant than when running an application on a standalone desktop computer.

Some applications require more security than others, playing Sudoku on a web application would cause little concern, but dealing with sensitive corporate formulas or accounting details in a web environment might be determined risky.

Cost

Over the life of the software use, web applications are typically significantly more expensive over time. Desktop applications are purchased outright and rarely is their a recurring fee for the software use. Some desktop applications do have maintenance fees or fee based upgrades associated with them, but rarely is there a subscription fee associated with the software's ongoing use.

Many corporate web applications use a different model, users typically are charged monthly service fee to operate the software.

Fees are considered "subscription fees". If you fail to renew your subscription you may be unable to access the data stored in the web application.

Connectivity

Web applications rely on persistent and unmanaged connectivity. If you do not have an Internet connection or if your host does not have Internet connectivity you cannot access the information. Critical applications or businesses that are time sensitive cannot risk denial of service attacks or power outages to interrupt their operations and access data that is sensitive.

Slower

Web applications that rely on the Internet to transfer data rather than a computer's local hard drive, may operate slower. The speed may also vary based on number of users ac-

cessing the application.

Backups & Ownership.

Regardless of the platform, companies need to be sure that their data is appropriately backed up. When using a web application that are hosted by a third party, companies should clearly determine who owns the data housed in the application, and be sure that privacy policies prevent that data from being used by the web host.

Ultimately the accessibility of web based applications make them very desirable. Web applications have some fundamental limitations in their functionality, and are better suited for specific tasks. Understanding the pro's and con's to each business model, will help users determine whether a desktop application or web application will better suit their needs.

Why Projects Fail

Computer projects fail when they do not meet the following criteria for success:

- It is delivered on time.
- It is on or under budget.
- The system works as required.

Only a few projects achieve all three. Many more are delivered which fail on one or more of these criteria, and a substantial number are cancelled having failed badly.

So what are the key factors for success? Organisations and individuals have studied a number of projects that have both succeeded and failed and some common factors emerge. A key finding is that there is no one overriding factor that causes project failure. A number of factors are involved in any particular project failure, some of which interact with each other. Here are some of the most important reasons for failure.

1. Lack of User Involvement

Lack of user involvement has proved fatal for many projects. Without user involvement

nobody in the business feels committed to a system, and can even be hostile to it. If a project is to be a success senior management and users need to be involved from the start, and continuously throughout the development. This requires time and effort, and when the people in a business are already stretched, finding time for a new project is not high on their priorities. Therefore senior management need to continuously support the project to make it clear to staff it is a [priority](#).

2. Long or Unrealistic Time Scales

Long timescales for a project have led to systems being delivered for products and services no longer in use by an organisation. The key recommendation is that project timescales should be short, which means that larger systems should be split into separate projects. There are always problems with this approach, but the benefits of doing so are considerable.

Many managers are well aware of the need for fast delivery, leading to the other problem

of unrealistic timescales. These are set without considering the volume of work that needs to be done to ensure delivery. As a result these systems are either delivered late or only have a fraction of the facilities that were asked for. The recommendation here is to review all [project plans](#) to see if they are realistic, and to challenge the participants to express any reservations they may have with it.

3. Poor or No Requirements

Many projects have high level, vague, and generally unhelpful [requirements](#). This has led to cases where the developers, having no input from the users, build what they believe is needed, without having any real knowledge of the business. Inevitably when the system is delivered business users say it does not do what they need it to. This is closely linked to lack of user involvement, but goes beyond it. Users must know what it is they want, and be able to specify it precisely. As non-IT specialists this means normally they need skills training.

4. Scope Creep

Scope is the overall view of what a system will deliver. [Scope creep](#) is the insidious growth in the scale of a system during the life of a project. As an example for a system which will hold customer records, it is then decided it will also deal with customer bills, then these bills will be provided on the Internet, and so on and so forth. All the functionality will have to be delivered at one time, therefore affecting time scales, and all will have to have detailed requirements. This is a management issue closely related to change control. Management must be realistic about what is it they want and when, and stick to it.

5. No Change Control System

Despite everything [businesses change](#), and change is happening at a faster rate than ever before. So it is not realistic to expect no change in requirements while a system is being built. However uncontrolled changes play havoc with a system under development and have caused many project failures.

This emphasises the advantages of shorter timescales and a phased approach to building systems, so that change has less chance to affect development. Nonetheless change must be managed like any other factor of business. The business must evaluate the effects of any changed requirements on the timescale, cost and [risk](#) of project. Change Management and its sister discipline of Configuration Management are skills that can be taught.

6. Poor Testing

The developers will do a great deal of [testing](#) during development, but eventually the users must run [acceptance tests](#) to see if the system meets the business requirements. However acceptance testing often fails to catch many faults before a system goes live because:

- Poor requirements which cannot be tested
- Poorly, or non planned tests meaning

that the system is not methodically checked

- Inadequately trained users who do not know what the purpose of testing is
- Inadequate time to perform tests as the project is late

Users, in order to build their confidence with a system, and to utilise their experience of the business, should do the acceptance testing. To do so they need good testable requirements, well designed and [planned](#) tests, be [adequately trained](#), and have sufficient time to achieve the testing objectives.

Conclusion

These six factors are not the only ones that affect the success or failure of a project, but in many studies and reports they appear near, or at the top of the list. They are all inter-linked, but as can be seen they are not technical issues, but management and training ones. This supports the idea that IT projects should be treated as [business projects](#).

Summer Workout Tips

Just because the temperature outside is 95 and it feels like 110 when you factor in the humidity doesn't mean it's a good day to skip your workout. You can ditch that excuse now and use these tips to get you through a hot summer day and still burn plenty of calories to keep your metabolism going.

Summer Exercise Tips:

- Hit the gym for an air-conditioned workout, followed by a quick shower for a refresher.
- Hit the pool for a comfortable water workout. You'll never even realize that you broke

a sweat, but you'll feel great afterwards.

- Take advantage of the hours around dusk and dawn for outdoor workouts. This is when the sun is down and so is the humidity.
- Pop an exercise DVD in for a great indoor workout.
- Exercise in short bursts with breaks in between to cool off.
- Drink more than the recommended 8 glasses of water per day.

• Don't eat a big meal before exercising, but don't workout on an empty stomach either. A light snack is all you need.

• Use a home gym. Walking the treadmill in an house with air conditioning and fans is much easier than walking outdoors on the hot days of summer.

If you do choose to workout in the heat make sure you bring plenty of water, include time to warm up and cool down and listen to your body.

Stop if you feel ill.

Contact CFUS (corporate)

The CFUS Corporation * 4859 West Slauson Ave. Suite 219 * Los Angeles, California 90056
323.298.8502—office * 310.388.5988—fax * info@cfus.com * http://www.cfus.com

Story Credits:

<http://searchwarp.com/swa43806.htm>
<http://www.coleyconsulting.co.uk/failure.htm>
http://weightloss.suite101.com/article.cfm/summer_workout_tips