



## **Getting Current on PeopleSoft: Upgrade vs. Reimplementation**

## Getting Current on PeopleSoft: Upgrade vs. Reimplementation

One of the biggest decisions that must be made in the support and evolution of a purchased software application is whether to upgrade or re-implement. This one decision has far ranging implications on the entire organization and can drive the technology direction for years to come. The key is to understand what is an upgrade versus a reimplementation.

### What Is An Upgrade?

The simplest definition of “software upgrade” is moving from one (older) version of software to another (newer) version. Upgrading an ERP system is not the same as upgrading office productivity tools or web browsers. If it was that straightforward, there would be no need for “consulting upgrade services,” “upgrade education services,” “upgrade labs,” and whitepapers on the subject of upgrading software. What, then, do we mean when we talk about a “PeopleSoft upgrade?”

### What Is A PeopleSoft Upgrade?

First, we need to further define “upgrade.” In August 2005 PeopleSoft Financials version 8.9 was released. This version introduced additional functionality to the product. As the previous release was version 8.8, this release was considered a “point release” of the application software. A significant shift in the product could be marked with a naming change (such as the upcoming “Project Fusion,” and Microsoft’s Windows XP to Windows Vista), or with a



change to the leading version number. For example, version 7.5x of PeopleSoft required the application to be installed on a PC. In version 8.0, for the most part, an end user required only a web browser to run the application. The introduction of the “PeopleSoft Internet Architecture” (PIA) marked a significant shift in the product’s architecture and in the user experience, and this shift was reflected in the naming convention.

Within the PeopleSoft ERP realm, not only are there new application software releases, which bring new functionality to the product, but there are also PeopleTools software releases, which update the underlying development tools used to build and run the application.

The final layer of complexity, in addition to “point releases” and major releases, is the ongoing maintenance to both application and architecture software. These “fixes” (or “patches”) are typically separated by product, and combined into “bundles” of fixes. Bundles are issued somewhat frequently for both the technology (PeopleTools bundles) and

application. At some point during the version's lifecycle, the bundles themselves are bundled into Service Packs.

### Defining The PeopleSoft Upgrade Process

It is important to note that, although an upgrade path may be supported (that is, upgrade scripts can be used to move from one version to another by using upgrade scripts), it may not be a *direct* upgrade path. In other words, going from "Point A" to "Point C" may require going from "Point A" to "Point B" and then to "Point C." Release notes (which describe the new functionality delivered in the release) only discuss the delta between the previous release...so the process of figuring out all the new features or changes requires reading through each set of release notes.

At any point in time, an organization may be faced with the need to apply bundles, Service Packs, a point release, and / or a major release. We do not want to minimize the complexity in applying bundles and Service Packs – these upgrades present their own unique challenges. For this discussion, we will address the considerations in moving between two or more point releases (version 8.0 to version 8.9), or in going to a new major release (version 7.5 to version 8.9).

### Reimplementation Considerations

Every PeopleSoft installation is unique. The decision to upgrade or reimplement is similarly

unique, and is made more so by the business requirements, IT infrastructure, budget considerations, and the organization's capacity for change. However, there are certain key factors that should influence the decision to reimplement, such as:



1. **Running An Unsupported Version:** In general, PeopleSoft maintains support for 5 years for a release (as noted above Oracle has announced some exceptions to this rule). After that, the release is retired, and support for fixes and upgrade scripts cease. The Oracle | PeopleSoft "Customer Connection" website will likely have the upgrade documents, but will not provide any assistance if issues are encountered. Significant architecture changes between versions may also prove difficult to manage via an unsupported upgrade.
2. **Running Multiple Versions:** The opposite problem of running an unsupported version is running more than one version in



production. This situation is actually somewhat common...a new version offers desired / required functionality, or perhaps a new module is introduced...and it is less expensive (up front) to simply install and configure the new software and build interfaces as required between the old and new. Once this precedent is set, an organization may find itself with more than two versions. Rather than run multiple upgrades and still be faced with different databases, a holistic solution is to implement the latest version of the software and convert the "silos."

3. **Business / Organization Changes:** Over time, an organization may grow in size (organically or through acquisition). It adapts to its existing marketplace by introducing new products and services. It expands into new territories, whether new regions of a country or across borders, to new countries. A PeopleSoft system not designed with currency considerations in mind will likely be unable to accommodate any level of

currency transaction processing. These changes themselves may put significant pressure on the existing PeopleSoft configuration. They may also drive organizational restructuring and reporting. This change alone may not require a reimplementation of the software. But if the initial configuration did not accommodate future growth, or if configuration maintenance has not been particularly rigorous, the organization may find itself constrained by the software, instead of enabled by it.

4. **The "Paved Cowpath:"** In the initial PeopleSoft implementation, was there a rigorous examination of existing business processes and how they compared to the processes embedded with the software? Was the existing Chart of Accounts streamlined, eliminating unnecessary values or reorganizing values to eliminate hard-coded logic (i.e., "the last 4 positions of the Account field are the Product")? Was a report inventory taken, to understand exactly what the previous system was reporting, or if anyone was actually looking at them? Was the PeopleSoft Query tool used to eliminate certain types of reports? In some instances, PeopleSoft may have replaced the "legacy" system in name only. This trap, while always present ("we have unique requirements" or "that is just the way it works"), was especially easy to fall into in the late 1990s, during the Y2K frenzy of implementations.



A “technology slam” may have solved the Y2K problem but further calcified the business processing.

5. **“The “Long Time Listener:”** On a talk show or web discussion group, have you ever heard the “long time listener, first time caller?” They have been faithful to the program, tuning in and enjoying the show, but haven’t been an active participant. In a sense, a long time PeopleSoft customer may faithfully upgrade, but has never “called in.” The processes and functionality initially implemented proceed unchanged. The organization may not be taking advantage of new functionality or (potentially) new modules. Worse, functionality that required a customization may have been introduced into the supported product. The analogy, while not completely accurate, is to upgrading your word processing software...how may new features in the latest version (or previous version?) of Microsoft Word have you really used? Implementing the latest version as a new system can force a focus on new processes and functionality that may add value to the organization.
6. **Technology Shift:** The forces that pressure an ERP system may be, as outlined in point 3, related to changes in the business or in the organization. Technology, certainly, does not stand still. Open source, not on the “radar screen” of many organizations even a few years ago, may now be a high priority. Organization changes may drive technology changes...the existing infrastructure may no longer scale to meet service level agreements. Perhaps a new database engine is in order. The “cans and wires” that supported the software can no longer do so. A script-based upgrade that swaps out one database for another can be completed, but does present a host of special challenges. When seeking to change aspects of the technology landscape, an organization should pay special attention to the other factors outlined in this paper...it may not take much to tip the decision to a reimplementation.
7. **Customizations:** One of PeopleSoft’s key selling points continues to be the relative ease of customizing the delivered software. If data entry pages can’t be tweaked via configuration (entry templates or the “customize page” hyperlink), it is a ‘snap’ to hide or add fields, change defaults, and so on. A new table can be defined and created in minutes, and its table maintenance page created in not much more time. Certainly, not every customization creates “bug fix” or

upgrade issues, but despite best efforts, even the “clone and modify” approach does not work. Here, the modifications within each object must be compared against the new object, to see if it can be directly re-applied. If not, and if the customization is still required (see point 5 above), a re-development effort will be required. The point of diminishing returns is reached quickly when trying to upgrade a fairly customized PeopleSoft application. Any decrease in time provided by the “automation” of a script-based upgrade is spent re-applying, re-architecting, or removing the customization.

\* \* \* \* \*

The decision to execute upgrade scripts against a PeopleSoft application, or to install a new release and treat the current version as a “legacy” system, depends on many factors. The rigor of the initial implementation, reengineering during previous upgrades, and the method of adapting new functionality or new modules is a consideration. The direction of the business, be it in the products and services offered or where they are offered, can point the way forward; in the same way, the infrastructure environment can open or close doors. A careful analysis of the costs and benefits to upgrade or to reimplement should consider these factors.



## About the Institute for Value Creation

Delivering exceptional results requires exceptional knowledge—of industries, of solutions, of market conditions, and of business processes. Impart Solutions's Institute for Value Creation (IVC), provides understanding and solutions to key management issues through IVC's series of research, events and whitepapers.

Contact [ivc@impartsolutions.com](mailto:ivc@impartsolutions.com) for questions or comments on this and other whitepapers and events produced by the Institute for Value Creation.

## Impart Solutions, Inc.

714 W. Burlington Ave.

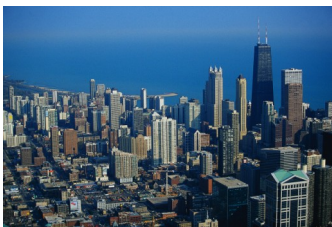
La Grange, IL 60525

(708) 354-8302

(312) 577-0978 fax

[www.impartsolutions.com](http://www.impartsolutions.com)

[info@impartsolutions.com](mailto:info@impartsolutions.com)



## About the Author

### David Aliosius

David Aliosius has been helping organizations meet their goals through the use of technology for over eighteen years. His diverse information technology experience includes designing and implementing both custom-developed and package software solutions. He has utilized this experience to support companies in many industries, including professional services, financial institutions, manufacturing, and transportation. Prior to joining Impart Solutions, Mr. Aliosius held a senior level position with a global consulting firm, focusing on Enterprise Resource Planning (ERP) solutions.