PeopleSoft Upgrade - The Lab Approach and Why We Did It That Way

By Brad McCann
Alfa Insurance

Introduction

In this presentation, we will explore how and why Alfa Insurance chose to use the lab approach for our HR upgrade while supplementing our internal resources with consultants to make it a huge success. Learn how we leveraged our partner to enhance our knowledge and how important it was for us to work with a partner who we knew would help us be successful. As always, planning and execution are important, but equally as important was our ability to adjust to the unexpected along the way.

Alfa Insurance is a regional multi-line insurance company headquartered in Montgomery, AL. We offer the standard property and casualty lines of insurance (Auto, Home, Fire, etc) as well as a variety of Life products and annuities. With about 2,700 employees spread across some 400 locations, our offices contain anywhere from two to 1,000+ people (the home office). Alfa's PeopleSoft systems are running on Windows servers accessing DB2 data on our z/OS mainframe. We utilize the PeopleSoft HR, Flex, Payroll, Benefits and Pension Admin modules as well as six of the eApps for employee and manager self-service.

Upgrade Objectives

Alfa was running the 8.3 version of PeopleSoft and needed to upgrade to remain on support so we could continue to receive tax updates and other system patches and fixes. We are also planning to remain on PeopleSoft applications until we have a better idea of the direction and technology stack used in the Fusion application, specifically as it relates to support for DB2 on z/OS. To that end, we wanted to move to a version that not only had a planned upgrade path to Fusion, but one that offered the ability to purchase extended support after the scheduled support expired. Our desire is to remain on this version for several years, until we can make an informed decision about Fusion.

Anticipated Challenges

Alfa has a team of developers and an administrator dedicated to nothing but PeopleSoft as well as shared access to server and DBA resources for maintaining our PeopleSoft systems, both HR and Financials. We also have about twelve people who work in our HR department. We always face challenges when undertaking a PeopleSoft upgrade. First, our internal resources must continue to support production as their primary duties even during an upgrade. Second, we know precious little about the newer version that we are upgrading to, which may have some significant changes that we need to understand. Third, it seems each version has two to three times as many DB2 tables as the previous version, so data verification, system sizing and tuning are a challenge for us.

After our initial planning, this upgrade introduced another challenge; the fact that we chose to perform a double upgrade from 8.3 to 8.9 to 9.0, which was bleeding edge, a place where Alfa rarely finds itself. Based on conversations we had with trusted consultant partners, we decided the risk of doing a double upgrade and going to the newest version was acceptable in order to get almost two additional years of support. This decision was dependent on Oracle getting the upgrade scripts for DB2 ready in time for us to stay on track since the scripts from 8.9 to 9.0 were not available at the time of our decision. What we decided was to start the process and once we had completed our initial 8.3 to 8.9 upgrade, we would make a final call on whether to stop at 8.9 or go on to 9.0 based on script availability. The scripts were available in time and so we went through with our plans for a double upgrade.

Partner Selection

As a public company going through Sarbanes-Oxley testing each quarter, we decided we must go live on the new system no later than November 1, 2007 or we would postpone our go live until the first quarter of 2008. Since we would be starting around March 1, 2007, that meant we had to perform a double upgrade to a new version in something less than eight months. While we felt that was possible, we actually had to start before the complete upgrade scripts for DB2 on z/OS were even available for the 8.9 to 9.0 portion of the upgrade, as previously noted.

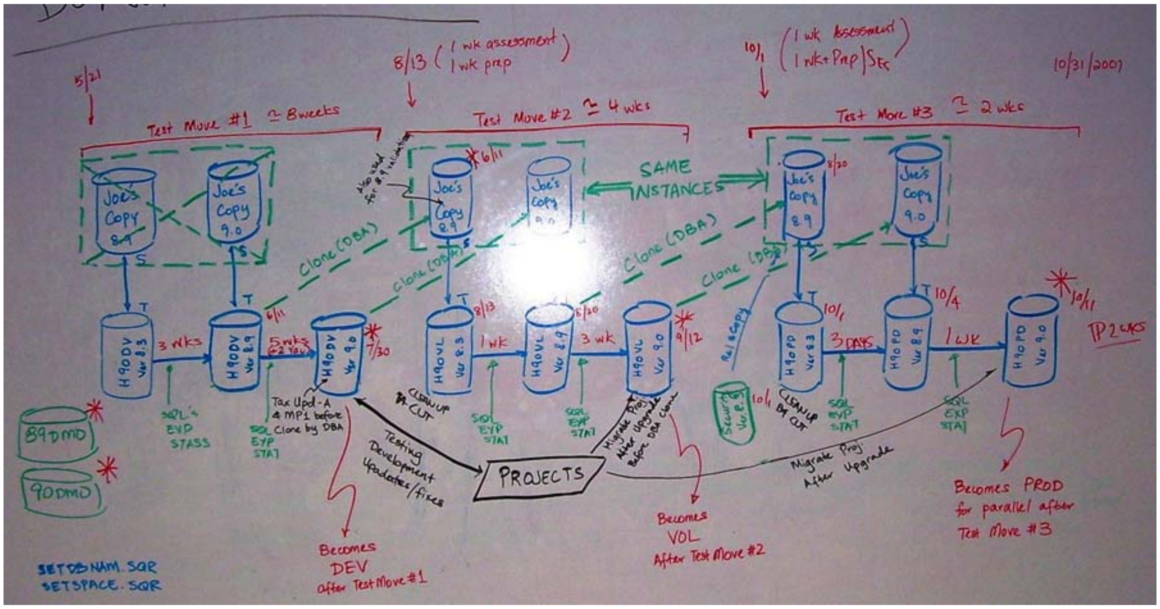
We knew that selecting the right partner would be crucial in order to give us the best chance to complete this double upgrade. Alfa needed a partner that could provide four major roles for us. Those roles were the technical upgrader, a functional/technical leader that could also co-manage the project, a DB2 DBA to assist our internal DBA in planning for the various instances required, and a very strong technical system resource that could work with our team to “pull back the covers” and help us learn and understand what was taking place and how to troubleshoot issues.

The most difficult resource for Alfa to find is a DB2 DBA resource that can really help us and is willing to follow Alfa’s procedures and standards. The other need that we really were looking for was a more complete knowledge transfer around the system workings, including assisting us in reviewing our logs and helping us understand the relationship between the messages we were seeing and the system configuration options selected during installation.

After talking to several potential partners, we selected System Efficiency, based in Richmond, VA.

Planning the Upgrade

The first step we took was a multi-day onsite planning meeting with the key resources on both sides. We had to understand all the instances required to complete this double upgrade that would allow us to complete the project in less than eight months as well as the timing of those instances and their relationships to each other. The result of that planning session was the following diagram which outlined the process.



We then translated that into a series of Visio diagrams where we further detailed the dates that we needed the associated servers and databases so our tech team could plan their work. This also helped us know where to take our tools exports, when to merge in projects of the retrofits we were making, and when to merge in our final security changes that had been taking place in production throughout the project.

Why the Lab

Based on our decision to upgrade from 8.3 to 9.0, we knew we had additional work, multiple verification steps, and increased risk on this project. All those translated to additional time that this project would take, so time became a very critical constraint since we were not sure what we would encounter with brand new upgrade scripts, data validation, etc.

Based on these factors, the lab approach made sense to use for several reasons. First, we could get started quickly since all we really needed to do was take a system backup (freeze) and provide an export to the consultant. Our partners were already prepared to perform the initial tools upgrade in their lab. Second, this gave us time to do additional detailed internal planning in parallel with the initial upgrade. Third, we had time to procure new hardware required for the upgrade. Finally, it gave us time to work out the validation and testing schedule for the upgrade and then tweak it around activities required to keep the business running … you know, critical activities such as payroll.

Remote Access Benefits

After our upgrader came onsite to complete the initial upgrade pass (data conversion), we decided to provide a VPN connection with full capabilities for him to work remotely. While not technically considered part of the “lab” approach, we thought of it that way because it was a way to allow him to accomplish his work without having to be onsite. This decision ultimately allowed us to complete the work just as efficiently, while giving us some options when the schedule and costs began to get out of line with our plan.

We were able to change our plans and have our upgrader work remote for an additional seven weeks, and the savings in flights, hotel, meals, and rental cars helped us cover the additional consulting hours needed when those exceeded our initial plan. Ultimately, we completed the project and went live on October 15, 2007, which was two weeks early and less than 1% over budget.

DB2 on z/OS Lessons Learned

As a DB2 on z/OS installation, there are a number of things we learned during the upgrade specific to our environment. Below is a list of those items. If you have questions about the details, I can put you in touch with our DBA who can explain in more detail what he encountered.

- Apply maintenance to OS and DB before upgrade
- Understand rules and limitations of new versions of DB2
- Indexes delivered with DEFINE NO can cause problems (see previous point)
- Need to have < 1000 tables per tablespace on DB2 in non-DMO environments if using DSN1COPY
- Run SET programs more often to preserve customizations/modifications
- Back up with multiple methods of recovery to have flexibility in recovery options
- Exploit DASD utilities like SNAP or Flashcopy
- Segregate live environments in separate DB2 subsystems if using SNAP
- Buy a ZIP specialty engine to reduce contention for mainframe core MIPS with other systems

Upgrader Tips

We used a consultant to perform the actual upgrade steps, which was more complicated than usual since this was a two step upgrade. We first had to perform the upgrade from 8.3 to 8.9 and then from 8.9 to 9.0. Below is a list of key tips learned by our upgrader. Again, if anyone needs more specifics on these items, we can get more details for you.

- Do not apply patches (other than “Required For Upgrades”) to the databases before running conversion script as the App Engines do not account for any object changes in these patches
- DB2 – pay attention to STOGROUP parameters in the export and import scripts, this is in addition to tablespaces
- In a double upgrade, two source databases are needed. In the 1st database, bring all customizations forward, then spend time on 2nd database compare
- Security – allocate at least 25% of project time for application, design and implementation
- During testing and/or go-live troubleshooting, 90% of issues and “bugs” are security related, that is where all troubleshooting should start
- Combine all post upgrade patches/bundles into one big project (as opposed to individual projects) and apply that project to database after Test Moves

Summary

We have performed three upgrades to our PeopleSoft systems (two for HR and one for Financials), and on both of the HR upgrades, we used a similar “lab” approach. It has worked well both times and it is something we would not hesitate to do again. It is my understanding that it would be even easier if we were an Oracle DB installation instead of DB2 on z/OS.