Upgrading a Customer Care and Billing System

An Oracle White Paper
December 2007

Over the entire life of your customer care and billing system, upgrades are a significant cost. You can vastly reduce those costs, however, by choosing a product whose architecture is specifically designed for easy upgrades.
WHY THINK ABOUT UPGRADES?


To keep pace, you'll need to upgrade your customer care and billing system. Vendors of packaged systems typically provide new software releases every 18 to 36 months. They add functions and take advantage of new technologies and industry standards.

There are almost always costs involved in putting the new release into production. The size of that cost depends on your software’s architecture. By choosing a customer care and billing system architected to support fast, easy, low-cost upgrades, you will reduce the negative impact upgrades can have on staff productivity and day-to-day operations. You will also reduce total costs of system ownership over time.

That’s why upgradeability should be a key decision factor when you select a new customer care and billing system.

UPGRADES REQUIRE THE RIGHT PRODUCT ARCHITECTURE

Custom vs. Packaged Systems

In the past, customer care and billing systems were custom-designed and custom-built. Costs were high. The systems were difficult to maintain and change. Migrating them to new hardware was difficult or impossible.

Far better are today’s packaged, vendor-supported customer care and billing systems. They generally arrive in your IT department as a series of programs you implement on your hardware. You hook the package up to other applications throughout your enterprise by using pre-packaged integration from the vendor or by developing unique links. You then configure and customize the package to your exact needs.
If You Customize the Core, “Upgrades” Turn into Difficult, Expensive Re-Implementations

To adapt most vendors’ software to your needs, you must modify the core programs provided. As a result, when the vendor sends you an “upgrade” you must make all those customizations again. They’re not preserved. The process is frequently a costly and time-consuming re-implementation and/or retrofit. In the most extreme cases, companies must hire systems integrators to design and implement elaborate upgrade strategies.

Costs for each re-implementation under this type of architecture can equal or exceed the cost of the original implementation.

Forcing New Functions Simultaneously on Staff Erodes Productivity

Typically, upgrades put a number of new functions at your staff’s disposal. Unfortunately, many vendors have an “all or nothing” approach to these new functions. Your staff must cope with all of them simultaneously. Confusion, errors, and low productivity are inevitable results.

Far better are upgrades that permit you to turn on individual functions as you need them and as your staff is ready to absorb new procedures and options. Staff members become skilled in the new top-priority applications before they address others.

The Architecture of a True Software Product Enables Fast, Easy Upgrades

To avoid unnecessary expense and system problems, you should select a customer care and billing system with an architecture that adapts easily to the inevitability of change over time. While offered by only a few vendors, this architecture has many advantages. It:

- Isolates core programs from any adaptations or extensions you make to the system.
- Links the core to other applications throughout the enterprise with prepackaged interfaces, configurable interface templates, or tailored interfaces that use industry standards. These dramatically reduce both implementation and upgrade time and cost.
- Uses a model-driven architecture, a rich set of extendable metadata, and soft tables. These permit easy and tailored system configuration that is preserved during subsequent upgrades.
- Rejects customization that breaks the upgrade path and raises costs exponentially over time.
- Recognizes architecturally that there are always unique needs that require more than just simple table configuration. It thus permits adaptation and
extension using configurable algorithms to augment core processing without breaking the upgrade path.

- Delivers—as part of the product, and with each new software release—the necessary tools to automate the upgrade. These tools substitute the new core for the old while preserving the integrity of the tailored algorithms, configurations, and metadata changes and extensions.

- Comes with testing scripts that ensure the system is working properly.

When it’s time to upgrade a true software product, you install the new release, perform the upgrade steps, and run the automated upgrade procedures. That’s all. You are now ready to begin your testing, training, and business-process reviews in preparation for moving the upgrade’s innovations into production.

### UPGRADE COSTS

When architecture promotes fast, easy upgrades, costs are dramatically lower.

While multi-party studies on customer-care upgrades are difficult to obtain, cost estimates of $5-10 (US) per customer are not unusual. The experience of Oracle Utilities clients using the easily upgraded Customer Care and Billing architecture, however, is that costs are far lower.

Upgrading a single component also shows major savings when you start with the right architecture.

In some rare cases, upgrades with even the best architecture may require minor work to update unique, tailored algorithms. But these upgrade costs are vastly lower than the costs involved when links and customizations modify the vendor-supplied software, requiring reimplementation and/or retrofitting of the released changes. Over the long lifespan of a typical installation, these cost savings really add up.

### HOW TO RECOGNIZE ARCHITECTURE THAT UPGRADES EASILY

The technical reviews you conduct during the selection process are rarely helpful in predicting the ease of future upgrades. Your best information will come via close questioning of vendors and their clients. You’ll need to ask vendors:

- What does the typical upgrade cost your clients per end-use customer? How much is their cost per customer to do the technical upgrade? To do the testing? To train staff? What does it cost to get the upgrade into production (both total cost and the percentage of initial implementation cost)?

- What is the average length of time your clients have been on their current version? (If the answer is more than 4 years, chances are that the software does not foster easy upgrades.)

- Are new versions of the software shipped with all the automated processes needed to perform the technical upgrade?
Can I typically receive the new version on Friday, do the technical upgrade of my test system over a two-day weekend, and begin testing on Monday?

What options exist for upgrading single components as opposed to entire systems?

How many upgrades have there been to the product? How many of your clients (and what percentage) have upgraded the product?

What is the typical time between signing an upgrade contract or receiving an upgrade and going live on the new release?

You'll also want to talk with companies currently using the product you're considering: While many of the questions are similar to the ones you've asked the vendor, it's helpful to examine upgrades within a specific context:

- Have you ever upgraded to a new major version or release of the software, or are you still on the version you went live with? How long have you been on your current version?

- If you have done a complete upgrade, or if you have a quote for an upgrade, what is the cost per end-use customer to do the technical upgrade? To do the testing? To train staff? What is the total cost of getting the upgrade into production? (Look for total costs of $2 (US) per customer or less.) How does that compare with the cost of the initial implementation and customizations?

- What new advantages did you experience after upgrading? Were you able to streamline processes? Use new functions? Implement new procedures?

- What was your experience with the processes needed to perform the technical upgrade? Could you perform the tasks in the time expected (for instance, over a 2-day weekend)? How much manual work was required to re-implement your “custom” changes?

- What has been your experience with software problems? Do you receive a “single fix” that is well documented? Easy to install? Quick to get into production?
MAXIMIZE YOUR INVESTMENT

Your new customer care and billing system can serve you well for many years—but only if it adapts to changes in your company, in technology, and in your customers’ needs. Upgrades are key to assuring system longevity.

To minimize total costs over the life of your system, choose a customer care and billing package architecture that assures fast, easy, low-cost upgrades from a vendor dedicated to helping you maintain staff productivity despite system and customer change.

“The figures now being released regarding [Oracle Utilities CC&B] upgrades mark a major turning point in the industry. In the past, upgrade costs of $5 - $8 per customer would have been considered very low-cost indeed; $20 and up would probably have been more realistic. We’ve had utility clients that have spent millions of dollars to upgrade systems with very little to show for it. Others have looked at the cost to upgrade systems installed as recently as 2000 and have instead elected to implement an entirely different vendor’s system.”

—Greg Galluzzi, TMG Consulting