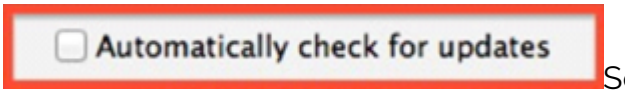
AUTOMATIC UPDATES FOR BMC DB2 SOLUTIONS (PART 1)

Nearly every device I own automatically pulls in software updates.

- My Mac, automatically updates OS x
- My pc, automatically updates the MS Office Apps
- My iPhone, automatically updates iOS

You get the idea. From laptops to cell phones, everything seems to have one of THESE:



So why does keeping mainframe software up to date seem to be so difficult? Especially since it is also possible to automate mainframe software updates – as we will see.

The highly manual and challenging aspects of the process often keep our usually happy mainframe customers from undertaking necessary upgrades. For example, almost every time a customer calls in with an issue, I must send an email informing them about far back they are on maintenance. This impacts how quickly we can jointly get to issue resolution. Customers calling me to be coached through new "whiz-bang" features they saw on one of BMCs YouTube videos, are often disappointed to find out that they cannot use these new features because they are not current on maintenance. (When I give bad news, I always make sure to include the requisite sad face 😟 in my responses).

Managing a mainframe product installation or upgrade can be a rather complex and arduous

process. You have to deal with SMPE to pull and apply maintenance PTFs. Some of these are fine, some are in error and nearly all have manual HOLDDATA instructions that need to be read, analyzed and carried out by you manually. The installed code base has to be in synch with the Db2 version and the Db2 repositories that the codebase operates against and manages. The code can't read a Db2 table if the table hasn't been created, and things might not perform well if an index is needed. But what if the index hasn't been created because someone missed the HOLDDATA instructions?

Let's pause and consider this complexity from two perspectives:

1. **Slow to upgrade**. If you're behind on maintenance, everyone suffers. The installer, usually under intense time pressures, will have to grab the PTFs and possibly open cases with the vendor after discussing the maintenance with their DBA. Often the installer plays the middle man getting information from their DBA about what the issue is and communicating it to vendor support to be analyzed. An awful lot of time is being expended in these three areas: the vendor, the installer and the DBA or user.
2. **Eager to upgrade**. The other view is this – "Am I being too aggressive with maintenance?" What if you make a change that detrimentally impacts your applications? Can this risk be mitigated with proper controls? Maybe, if you aren't overly aggressive in the PTF capture part of the picture. Or, you can be more aggressive accepting and deploying maintenance into lower life cycle locations, and perhaps less aggressive deploying into critical production areas. Though this is hardly ideal and can increase risks. With automatic updates, one can also build in automatic fall backs supporting a more aggressive maintenance schedule while at the same time lowering the associated risks.

All in all, automatic updates can be a big win for a customer's product installations; your code will perform better, and your customer satisfaction will be higher as well. The installer will have significantly less to do, the DBA will have a much more robust set of products to work with and exploit, and the business will run more smoothly and productively having the latest code and latest features operating.

At BMC, we have a process that enables a software deployment to run in a quarter hour outage, and fallback (if needed) runs in minutes.

Next time, we will look at some of the components for putting **Automatic Maintenance** into action. We will leverage the methodology BMC has put in place with version 12.1 of our Database Management Solutions for Db2 to deal with IBMs **Continuous Delivery**.