CHAPTER 9

When Should You Upgrade Your BAS?

What's in this Chapter

When should a building automation system be upgraded?

A lot of folks I meet have this exact question. The challenge with upgrading a BAS is figuring out exactly where the tipping point is.

What do I mean by the tipping point?

The tipping point is when an existing BAS becomes significantly more expensive to maintain than the cost of upgrading. This brings me to my next question.

How do you know when you have reached a point where your system needs to be upgraded?

In this chapter, I am going to discuss both major upgrades and the often ignored minor upgrades. I am also going to talk through firmware (that is a fancy way of saying controller software) upgrades.

So here is what you are going to learn about in this chapter:

- Making the decision
  - How do you know when your BAS is too old?
  - What parts of your BAS should you upgrade?
  - Monetizing the risk of not upgrading
- Planning the upgrade
  - How do you plan for an upgrade?
  - What can you expect on the day of the upgrade?
- Advanced topics
  - How do you handle multi-vendor upgrades?
  - How do you maintain integrations when you upgrade?

Ok, are you ready to answer the question that has been plaguing both owners and contractors for decades?

It's an ambitious goal, but by the end of this chapter you will understand:

- How to make the upgrade decision
- How to plan upgrades
- How to handle complex upgrades

Alright, let's dive in!

Making the decision

Let's face it upgrades are scary. Whether you are the customer or the contractor, there's a lot of risk in upgrades. Often, even the customer doesn't know what is in their building.

So there you are, trying to figure out how to upgrade a system, while also trying to convince the customer (sometimes this is internal customers) that this upgrade is necessary.

And that is why this section is first.

At the end of the day, your main goal should be to understand why the upgrade is happening and the benefit that you or your customer will see because of this upgrade.

I argue if there is no benefit to an upgrade why upgrade?

Don't worry. I'm going to cover that question as well.

So let's start off with the age-old question, how do you know when your BAS is too old?

How do you know when your BAS is too old?

I've built my whole career around upgrading and integrating building automation systems. It never fails to amaze me some of the ways my customers were able to keep their systems running long beyond their prime.

I still remember the first time I got shocked by 120V, trying to retrofit a pneumatic system that was on its last legs.

No matter what I tried, I just couldn't get the fan on this unit to turn off. So there I am ripping out wires and cutting out pneumatic tubing, and I see this alligator clip holding two wires together. Because the wires were so small, I thought that they were communication trunks.

So what I do?

Well, I just reached out and grabbed that alligator clip and pulled it off the wire. I then proceeded to have my arm thrown back into sheet metal enclosure that I was standing inside of.

Man, that sucked.

And that is how I learned that this jury-rigged building automation system had outlived its useful life.

When you have to start alligator clipping wires together or fusing pneumatic tubing together just to get things to work, it might be time to consider replacing that system.

But I know you all, you're not get a let me out of answering this question that easily.

No, no, you all want a checklist on how to determine when your BAS to is too old.

Well, ask, and you shall receive. Below are the four questions I ask to determine if the BAS is too old:

1. Can I no longer buy parts for the building automation system?
2. Is the building automation system manufacturer out of business?
3. Can I no longer find anyone who knows how to support this building automation system?
4. Am I spending more to maintain my existing system than it would cost to upgrade the system?

Whether you are a contractor or an owner you should stop and ask yourself these questions.

You see, upgrading a building automation system should not be an emotional decision. In my experience, the decision to upgrade a BAS is made based on the answers to these four questions. The answers to these questions will help you determine if a building automation system is no longer viable.

What parts of your BAS should you upgrade?

At this point, you probably have one of the following questions.

The questions go something like this.

What parts of the building automation system should I upgrade?

I mean, surely you don't expect me to upgrade the entire thing?

Or

How do I know if the entire building automation system needs to be upgraded?

It depends on how you answered the questions from the previous section.

For example, if you have a BACnet MS/TP communications trunk and the field controllers can no longer be bought. Then it makes sense to upgrade your field controllers with new field controllers and leave your supervisory controllers and servers in place.

The reason to upgrade the field controllers is because BACnet MS/TP is a common protocol and there are tons of controllers that support it.

However, if the BAS servers and supervisory controllers can no longer be bought or they use a proprietary protocol, it may not be possible to replace the field controllers.

Therefore, this problem should be approached on a case-by-case basis. There are three parts of the building automation system that an upgrade project typically focuses on. You may recall these from Chapter 2:

- Building automation server
- Supervisory device
- Field controller

Here are four questions that you can ask to determine which parts of the BAS you should upgrade.

As you ask these questions, it is important to start with the building automation server and work your way down to the field controllers. As you move through each layer continue to ask yourself or your customer the following four questions:

1. Does this software support open protocols?
2. Are these open protocols present in products on the market?
3. Can I still find people who will service this system?
4. Can I still purchase the system (this is mainly focused on the supervisory devices and field controllers)?

As I said, you will continue to ask these four questions as you move down from the supervisory device to the field controller.

Whenever you discover that the answer to one of these questions is no, you need to make a decision about whether or not you want to take the risk of continuing to use that system.

Monetizing the risk of not upgrading

How do you determine the risk of not upgrading?

In chapter 12 I will describe how to determine a return on investment (ROI) for service activities. I've included the formula from chapter 12 below.

$$ROI = \frac{(\text{Gain from Investment} - \text{Cost of Investment})}{\text{Cost of Investment}}$$

You can follow the ROI equation and process from chapter 12 to determine the return on investment for upgrading, with some minor tweaks:

- First, you need to determine the cost of downtime. You do this by looking at what areas and functions of you or your customer's building and business, could be impacted by system failures
- Then, you will determine the cost of upgrading the system(s)
- Next, you will use the ROI formula to determine if the ROI justifies the cost of the upgrade
- If the cost of upgrading the systems meets your criteria, then you can proceed with the upgrade.

Planning the upgrade

At this point, you should have answers to the economic, technical, and operational questions related to this upgrade. You should also have evaluated the ROI for this project and decided that an upgrade is needed.

Okay, now what?

I've met a lot of folks who get to this point and freeze up because doing an upgrade can be pretty darn scary. I remember the first upgrade I ever did, and it didn't go well.

I went to a Native American hospital about three hours north of Seattle. I was supposed to replace a Tridium LON supervisory device with a new supervisory device and about a dozen new programmable field controllers.

Well right off the bat I set myself up for failure by not grabbing a copy of the database. My thinking at the time was that since I was only adding some new controllers, I wouldn't need a backup of the database. (Hopefully, you can read the sarcasm in my words) …

After going and taking the Tridium device off the wall and putting my supervisory device on the wall, I proceeded to wipe out the LON database completely. This led to the extremely fun process of spending the next three days at this hospital rebuilding the database from scratch.

To this day I'm still amazed that I didn't get fired, but in my defense, I've heard much worse stories than mine. The point is, I didn't have a plan, and that was pretty dumb.

But, you will have a plan, because I'm about to give you a step-by-step process for creating one.

How do you plan for an upgrade?

Yes, I'm talking to you, you've got to have a plan!

What follows, is a step-by-step plan that will help you to avoid the mistakes and pain that I experienced in my career.

You ready?

There are five steps, and if you follow them, they will make your upgrade projects much less painful.

Step 1: Determine the scope of your upgrade

What are you going upgrade?

Are you going to upgrade all the controllers in a building?

Just some?

Will you need to have a new communications trunk?

When you approach an upgrade, you will need to consider, the physical, logical, and operational impacts to a system. You also need to consider if your upgrade has the potential of impacting a part of that system.

I like to check the following areas:

- Will this impact the database?
- Will I only be replacing a couple of controllers on a communications trunk?
- Will I have communication speeds or protocols that could conflict?
- Will this new system need to work with the old system?
- Will I need to manage multiple user databases?
- Are there sequences of operations that are dependent on points from the controllers I'm upgrading?

This is not an exhaustive list, and it really will depend on what you are upgrading, but the list above should give you a head start.

The important thing is to check the physical, logical, and operational parts of the system(s) you are upgrading.

Step 2: Pick the order of systems you will upgrade

Now I realize you won't always start with a specific device. In some cases, you'll start with the supervisory device and other situations you may start with a field controller. The point here is not what device you start with. Rather it is the order in which you proceed to perform your upgrade.

As I see it, there are three schools of thought in when it comes to planning out an upgrade.

Method #1: Start with the field controller

In **method one** you start with the field controller and then gradually work your way up to the supervisory device. The benefits of this approach are that you only have to build the new database once. The cons to this approach are that you will have to go and run your systems in hand (meaning in manual override) until all of the controllers are upgraded.

Method #2: Start with the server or supervisory device

Method 2 starts with the server or supervisory device and flows down to the controller. The thought process is that you will go and stand up the new supervisory device with all of your field controllers in hand. Next, you'll start replacing the old field controllers, and as you replace the old field controllers, you will map your new field controllers into the new supervisory device.

Method #3: The hybrid approach

Method #3 is the hybrid approach. In this approach, you will take a parallel path where you are upgrading the supervisory device and the controllers at the same time. This approach has the benefit of being faster and minimizing the time systems are in hand. However, this approach is more complex and requires significant planning and expertise.

Step 3: Plan your rollback strategy

Okay, let's say your whole plan goes to crap what you do?

If all the sudden the controllers don't work or you somehow missed a linkage point between controllers, how can you get everything back to operational as possible?

In the IT world, this is called a business continuity or disaster recovery plan. Quite simply, the continuity strategy will detail out each change and the rollback method for that change.

For example, changing out a supervisory controller would be an individual change.

The rollback method for that change would be to reinstall the old supervisory controller.

In this step, you need to lay out each change and the appropriate rollback method for each change.

Step 4: Plan out your project schedule

Ok, now I'm going back to project management basics.

Quite simply you need to identify each task and the prerequisites for each task. Once this is done you need to assign personnel and timing to the tasks and work your way backward from the targeted completion date to the start date.

I'm not going to teach you project management, and resource mapping in this book. Just realize that you need to plan out your project just like any other project.

Step 5: Define your commissioning or startup process

Alright, the final step.

In this step, you will detail out how you will validate that the new system is functional. To do this, you will need to create a series of functional tests where you will go and check that the sequences and systems are operating as expected.

There are three ways to do this:

- You can spot check a subset of the devices that were upgraded
- You can hire a commissioning agent, which I covered in chapter 8, to validate that the new systems operate as designed.
- You can self-execute a full functional check on the systems.

I do not recommend method #3, as folks tend to get busy and never complete the functional testing.

How to execute an upgrade project

Okay, so now that you feel comfortable with planning an upgrade project it's time to talk through how to execute an upgrade project.

Now, I want to be clear that no execution plan will ever be 100% accurate.

Therefore, I do encourage you, to customize the 12 steps I am about to walk you through to fit your project.

With that being said, what I am covering in this section will form the bedrock of any upgrade project execution plan.

Based on my experience, there are 12 steps that you will want to take in order to execute your upgrade project effectively.

These 12 steps are:

Step 1: Decide on what day you will complete the project

I know this step seems obvious. Trust me. You'd be surprised how many upgrade projects are started with no defined completion date.

Often the owner, who may have never been involved in an upgrade project, doesn't feel comfortable asking for a date because they don't understand what is involved.

Owners, it's ok if you don't know!

You don't have to understand what is involved in asking for a date.

And contractors… pick a date!

Yes, I know things change and stuff pops up. As you move through these steps, you might find out things that make you adjust your date.

DON'T WORRY

There's an easy way to handle that, simply state something in your contract that says "*your current competition date is based on the information you have gathered so far and that the date is subject to change based on site conditions*."

Once again, I'm not a contract attorney so before using any specific contractual language make sure you get a legal expert to evaluate your contracts.

Step 2: Verify the job site systems and applications

If you have been following along up to this point you should have a list of the systems, you will be upgrading. Now you need to verify that the list of systems and applications that you will be upgrading is accurate.

This is a very simple step, but it can be time-consuming, so budget accordingly!

What you want to do, is to go to each system and verify the following:

- The location of the system
- The location of the systems manual override (if one exists)
- Any documentation related to the system
- Any code related to the system (if you are doing a software upgrade)
- The electrical and mechanical systems that supply or are supplied by the system

As you may imagine it can take a lot of effort to go and identify this information.

You may be tempted to skip this step. After all, you are going to be touching the systems anyway so why would you need to get all this information?

I'm here to caution you to avoid the temptation to skip this step.

I can speak from personal experience; any time I have skipped the information gathering portion of an upgrade I have ended up regretting it.

This regret can manifest itself in a variety of forms. One form of regret was labor that was unaccounted for due to the difficulty to access the system.

Another consequence of not properly collecting and verifying information is the additional costs related to not being able to upgrade a system due to missing software or configuration tools.

To make a long story short, make sure you verify the information I listed above before upgrading a system.

Step 3: Ensure you have the access you need

Access is critical for most upgrade projects.

But access to what?

I mean, we've already gone and identified the location, documentation, and code for the systems.

What else could you possibly need?

Well here is one huge area that tends to be missed. This area is the area of credentials.

What are credentials you ask?

Credentials, also known as usernames and passwords, allow you to access the applications or software for the systems that you're upgrading. Imagine that you are doing an upgrade to a system that contains programming to control other systems.

Would you just go and replace this system without understanding the code inside it?

Imagine the potential damage you could do if you just took wild guesses as to how the building automation system is performing its discharge air reset, optimal start, and zone averaging sequences?

I think you would agree that the potential damage you could cause to your systems is quite high if you don't understand what they are currently doing.

But you can't understand what your systems are doing if you can't access the systems.

You don't want to find out, two weeks into a project, that the systems manufacturer will not provide you access to the system and you have to guess as to how it functions.

Don't get stuck in this situation, make sure that you have the access you need before executing your upgrade project.

Step 4: Identify the systems that will be affected

But Phil, I've already gone and identified the systems that will be affected.

Did you?

Are you sure?

Often time's folks will only look at the systems they're upgrading, and they won't look at the systems connected to the systems they are upgrading.

For example, maybe you're upgrading the controls in a central plant. That's great because there are some awesome control sequences for central plants that have come out in the past couple years.

However, if you don't think about the systems that tie into and use the central plant you could end up with significant impacts on those systems.

Maybe you have a secondary chilled water loop that flows into another part of the campus. And this loop has some code that averages the valve position of your air handlers and resets a set point based on the average valve position.

Now imagine that you didn't know that this code existed. Because of this, the person who goes and writes the new code for the central plant creates a different way of controlling the secondary cooling loop.

Could this introduce problems?

Possibly.

Do you want to find out after the projects done?

Probably not.

So what can you do to avoid this scenario?

Well, I recommend that you go and identify the systems that will be affected or tie into the system you're upgrading.

Once you've identified these systems, you need to figure out what aspect of the system could be affected by the system you're upgrading. Once this is done you need to put a plan in place to ensure that the system(s) is not being affected by the upgrade.

Step 5: Decide on your upgrade strategy

In step two of the upgrade planning process, I described the different methods you can take when you perform an upgrade.

It's at this point that you need to select the method you will use. You will use that method to determine the people, tasks, and timing you need to put in your execution plan.

Step 6: Determine the people or groups that will be involved

This step is pretty basic.

By this point you should have a list with the following information:

- The systems you will be upgrading
- The systems you need access to
- The systems that will be affected by the upgrade

Based on this information, you will need to determine the people and or functional groups that need to be involved in the upgrade.

I recommend that you break this list of people into three groups. These groups are:

- Advise
- Approve
- Contribute

Advise

People who are in the advise group, simply need to be advised on any potential impact to their workspace or their systems.

An example of this would be notifying the head nurse when you're about to do an upgrade in a patient room.

Approve

People in the approve group will approve the actions you're taking and the allocation of resources to your upgrade project.

Contribute

People who are in the contribute group will be utilized to assist you in performing the upgrade. An example of this would be a facilities team member who needs to put a specific piece of equipment in hand for the upgrade to be executed.

Step 7: Write out your upgrade plan

At this point, you should have everything you need to write out your upgrade plan.

But if you've never written out an upgrade plan how do you do that?

In the bullets below I have provided you the eight areas that you will need to address to form your upgrade plan.

- Upgrade scope:
- Point of contact:
- System(s) to be upgraded:
- System(s) affected:
- Upgrade steps:
- Upgrade period:
- Completion date:
- Rollback plan:

You need to fill out each one these areas with the information that you gathered in the previous steps. You will need to complete these areas for each individual system.

Once you have these areas completed, you simply print out this form and provide it to the customer and post it in the area where you're working.

Step 8: Put the systems in hand

In step seven, you detailed out your upgrade plan.

One of the areas you filled out in this plan was the upgrade steps. One of the steps you should have, depending on the type of upgrade you're doing, is to put your system(s) in hand.

I highlighted this step because I have been on upgrade projects where the controls personnel did not put the system in hand. Because of this, the central cooling plant began to lose control, and this resulted in a loss of cooling to one of the buildings on a college campus.

Upgrades are complicated enough. You don't want to add further complications by leaving your systems in automatic control.

Step 9: Execute your plan

This is the moment you've been waiting for.

All of your planning, preparation, and research has led up to this moment. Now all that is left is for you to execute the plan you created in step seven.

There are a few important things to remember when executing an upgrade plan. These things are:

- Keep a copy of your upgrade plan near where the upgrade is taking place
- Practice your rollback scenario (if you have one) before executing the upgrade
- If appropriate put your systems in hand before upgrading them
- Notify the appropriate personnel when you begin the upgrade when you finish the upgrade, and if something goes wrong during the upgrade

Step 10: Document your changes

After your upgrade is complete, you will have to provide updated documentation.

This step shows how the upgraded system works and details out any changes that were made. This will assist with servicing the system in the future.

Specifically, you're looking for:

- Sequence of operations
- Schematics
- Wiring and flow diagrams
- Bill of materials
- Cut sheets on any new material

Step 11: Verify proper operation

Once the upgrade is complete, you have two more steps before you're done.

Depending on the kind of upgrade you performed, you may have changed or replicated the previous sequence of operations. It is important that you validate that the new or previous sequence of operations function as designed.

This is a pretty straightforward process where you do the following:

- In the case of a controller upgrade, verify all new and existing inputs and outputs still work
- If you were performing a supervisory device upgrade, ensure that your communication trunks still work and that you can still communicate with your controllers. Also verify that any, graphics, trends, alarms, and custom logic still functions as designed
- If you are performing a server upgrade, then verify that you can access the server, run your applications, and perform any server specific tasks that are required

Step 12: Check the opposite season control

This task does not have to be done on every upgrade project.

However, I encourage you to make sure that the systems you're upgrading do not have a different sequence of operations based on the time of year or season of the year.

An example of this would be a two-pipe system that has a summer-winter changeover, to switch between cold and hot water.

If a system does have a different sequence based on the time year, then you should plan to perform an additional functional test to verify that the upgraded systems function properly.

Advanced Topics

What I have covered so far will cover 95% of the upgrade projects you will be involved in.

However, there are two of aspects of upgrades that I want to cover in further detail.

Now I caution you. These are advanced topics, and it's perfectly fine if you simply move on to the next chapter.

In this section I'm going to cover:

- How to handle multivendor upgrades
- How to maintain integrations when upgrades are performed

How do you handle multi-vendor upgrades?

The reality is some upgrades will involve multiple different building automation vendors.

I experienced this back in Dallas, Texas, when I was working on upgrading the control of a central plant that was tied into another building automation system. What made this even more difficult was that building automation system had been purchased by another company and was no longer supported.

In order to support this new central plant control sequence, I had to make changes in the existing third-party system.

In this section, I will tell you how I addressed that and give you a step-by-step process for handling multivendor upgrades.

Much of what we cover in this section will be similar to our previous steps around planning and executing upgrade project. I am going to call out the differences rather than running through the entire process in full again.

There are three main areas you need to focus on when doing multi-vendor upgrades. These areas are:

- Identify the third-party systems that will need to be upgraded
- Ensure that you have access to the third-party systems
- Hire or contract someone who understands the third-party system

Identify the third-party systems that will need to be upgraded

You may recall that in step two and four of the upgrade project execution section I detailed out how to identify the systems that will be upgraded and affected by the upgrade. When dealing with a third-party system you simply need to perform the same tasks on the third-party system.

The only issue with this is you may not have the ability to access the third-party system or the expertise to work on the third-party system.

Ensure that you have access to the third-party systems

There are two ways to ensure that you have access to the third-party systems you're going to be upgrading.

The first way is to hire or contract someone from a third-party system to ensure that you have access to and the software for the system. This will allow you to perform your upgrades.

The second way is to work with the customer or your internal team (if you are the customer) to utilize personnel who are familiar with the system to determine whether you have access to the system.

Hire or contract someone who understands the third-party system

Okay, I've said several times now that you need to hire or contract someone who understands the third-party system but how exactly do you do this?

This is probably one of the most difficult areas when performing upgrades that involve multiple different vendors. The reason behind this is that upgrades are inherently risky, and the assignment of this risk is often disputed, especially between competing vendors.

So how can you go and hire someone, especially if that someone is your competition?

The best method I have found is to work with the owner. Obviously, if you are the owner, then just follow the steps below.

Step 1: Identify the people with the capabilities you need

This is quite simple.

You simply need to, identify the tasks that need to be done on the third-party systems and then based on that information you will need to contract the appropriate people.

Step 2: Define the scope that these people are going to perform

Before contracting these people, you will want to design a clear scope around what they will be doing. In the scope you will want to detail out:

- What systems they will be working on
- What tasks that will be performing
- When do these tasks need to be complete
- Who is responsible for the tasks
- The communication and escalation of issues related to the tasks
- The result or finished product that is expected after the tasks are done

Step 3: Contract the people to perform the work

How you contract people will depend on your organization.

The important thing is to have a separate contract for any work that involves a third-party system. The reason behind this is that the terms and conditions of the scope of work can be quite different from what you or your contractor has been hired to do.

I am not a legal expert, so I encourage you to seek out legal or professional advice related to the formation of contracts and subcontracts.

How do you maintain integrations when you upgrade?

While I will cover this topic in much greater detail in chapter 15, I still wanted to take a brief moment to mention a couple of key things.

First, you need to make sure, that you follow step 10 of the upgrade execution plan process. If you recall, step ten is where you document the changes to the system or systems when you're complete with the upgrade process.

As you may recall, the information you need to gather after the upgrade process is complete is:

- Sequence of operations
- Schematics
- Wiring and flow diagrams
- Bill of materials
- Cut sheets on any new material

But you need to gather couple more things to ensure that you have the proper information to maintain your integrations.

First off, before executing the project, you need to gather the physical and logical diagrams that detail out the existing integrations.

Next, you need to determine how, if at all, the integration(s) will change.

Finally, you need to test the integration(s), to verify that the integration(s) still function as designed after the upgrade.

As I mentioned, I will discuss integration in much greater detail in Chapter 15.

Chapter 9 Quick Summary

Upgrades, upgrades, upgrades...

This is one of the toughest topics for folks, and I understand why. You have so many competing emotions when you are going down the upgrade path.

Is the BAS contractor trying to sell me something I don't need?

How can I convince the customer that not upgrading is a very bad choice?

How can I upgrade this site that the customer has let languish for 30 years?

All of these questions strike at the heart of what makes upgrades challenging…

In this chapter, my goal was to make this scary area a little "less scary" for you.

I walked you through how to make a decision when it comes to upgrading. I showed you how to answer the questions around when is your BAS to old and how to identify what parts of a BAS you should upgrade.

I also taught you how to monetize your upgrades as well.

I then pivoted to the operational side of upgrades where I walked you through my step-by-step process for performing upgrades. I even showed you what to expect on the day of the upgrade.

Finally, I took you through some advanced topics like multi-vendor upgrades and upgrading integrated systems.

All-in-all I gave you a pretty solid framework to build off of for your upgrade projects.

And that completes section 2. Section 2 focused on the art of procuring, building, and upgrading BAS. In section 3 I'm going to take you through how to support the BAS you have.

I'm going to give you several step-by-step processes to guide you through managing your BAS!