CDBG-DR and CDBG-MIT Grantee-Led Sessions

6: Information System Evolution: Key Elements Added Over Time

Wednesday, March 31, 2021

Olivia Healey: Hi, guys, I'm Olivia. And if this is your first session, nice to meet you. If not, you're probably getting sick of me introducing myself. But key reason being that these do get recorded and you can use them as a resource down the road. And very beneficial tool for all of the sessions within the series and this session itself.

I'm accompanied by my co-host, Scott Ledford, and we'll take you through today's session. The sessions run with the session materials first, and then we'll follow them by Q&A. We have Q&As that were submitted in advance through our questionnaire. And then any questions that you guys submit throughout the session if a slide inspires you or you want to do it all at the end, make sure you get those questions submitted into the question and answer.

With that being said, let me introduce our speakers today. We have James, who's worked in a project management capacity for the state of New Jersey since 2007. James was offered an opportunity to lead the technical direction of the Disaster Recovery and Mitigation Division responsible for overseeing Sandy CDBG-DR funding where he's responsible for developing IT budgets, forecasts, negotiating contracts, managing vendors and in-house staff.

James also manages IT team members, ensuring day-to-day operations of the division are uninterrupted while planning for over 30 software and hardware upgrades each month. He works closely with the finance team to ensure systems provide accurate accountability while adhering to the rigorous standards of HUD's DRGR systems. And as of 2017, James began overseeing the PMO functions in addition to several capital projects underway simultaneously at all times.

Scott Ledford: Hi again, everybody. This is Scott Ledford and I have the pleasure of introducing Sam today. Sam has worked in state government his whole professional career. Beginning in 2010, he worked at the governor's office in a variety of roles. In 2014, Sam moved over to the New Jersey Department of Community Affairs. Since that time, he has held several leadership roles at the department, including his current role as the Director of the Disaster Recovery and Mitigation Division. In this capacity, he manages day-to-day operations of all Sandy recovery activities, as well as several other pre-disaster mitigation efforts. Without further ado, we'll hand it over to James to start the presentation.

James Shuster: Thank you, Olivia and Scott. First off, I'd like to thank you guys along with HUD for bringing us all together. Setting like these really help us all come together and understand the challenges that are sometimes unique to our circumstances. But also, chances are we share the majority of the issues that we're going to be discussing here today. So again, thank you for the opportunity for bringing us together here.

So hopefully that slide did change. Okay, so today we're going to be focusing on three main groups of topics, first of which is going to be priorities, next of which is flexibility, and followed by key elements that you're going to be building over time.

[Inaudible] priorities, it's important to realize that everybody has bottlenecks. It doesn't matter how many resources you end up throwing at the problem, how much money we throw at the problem, you're never going to be able to accomplish everything that needs to be done in the time that it needs to be done, especially in the unique circumstances surrounding disaster

recovery, where there's a great deal of upfront effort associated with getting the money out the door while simultaneously building the controls in place to ensure that the money isn't going to the wrong people.

How do you go about mitigating that? Effectively having the wherewithal of the organization as a whole and trying to stay ahead of critical components across all the programs that you're funding, and your program is going to be overseeing.

Similar to the next one here, staying ahead of program execution. What does this mean? Basically, identifying the bottlenecks within your system, where if you don't have them developed in time, somebody else is going to be doing something they shouldn't be doing or the program as a whole comes to a screeching halt.

Again, you need to have wherewithal of every program that's being run and what those bottlenecks are going to be for your specific program. Finally, be willing to push back on requests. Some of this is just basic project management. Specifically here, you're going to get requests sometimes five, six, seven times a day from various people that are requesting things that are important to them. And there's no argument that they are important to them. And sure, they are potentially feasible to the organization as a whole.

But at the end of the day, you're going to need to start making priorities about disappointing some people, and in some cases, some fairly high-up people in the food chain to make sure that the systems are continuously staying ahead of the organizational.

Building flexibility. This has a variety of different meanings, depending on your organization and circumstance. But basically, this is implying that when you are going to go and do something, having the wherewithal to understand when things are going to change and what is likely to change, and how they are going to change.

First item we got here is thinking big picture. Again, thinking across the entire organization. Who are the players? How often have the players changed or are the people that you're talking to able to make the final decision when you need a decision made? Or is it likely to, again, change as additional people get brought on board?

Building flexibly serves two main objectives. It allows you to change the system quickly, and it's also very expensive if you find that you have frequent changes that are required. An example of this is we recently introduced a new type of calculator to our system for grant awards. This calculator is completely database driven. The calculations are database driven behind the scenes. So when we need to introduce an element to the calculator or revise the version of the calculator, it can be done in a matter of minutes.

It reflects it all over reports and on the user interface, where before this new version of the calculator was rolled out, we would be taking three and four weeks to revise the calculator. And we have to consider how each of those changes to the calculator will not just impact new awards going forward, but also how they impacted previous awards that are already out there. Because if you told somebody they're going to be entitled to a certain amount of money and you're changing

an element of the calculator, that doesn't necessarily mean you can go back and change an agreement that's already in writing and signed.

Other factors to consider here. Just long story short, building flexibility is not always right. It does cost more and it does take longer. There are times when the right solution is to just put a field in the system or to develop a report and its face value and just move on to the next task.

So over the next several slides here, what we're going to be talking about is effectively a road map on how you build a system for disaster recovery. Honestly, this is something I wish I would have had when I started working here, and hopefully this benefits some of you, as well.

So we're going to start off on day one. This is effectively the day that you have access to funding. People are still sorting themselves out, they're figuring out what everybody is going to be doing. While they're doing that, you have an opportunity to get the systems in a good position. That's starting off by identifying infrastructure and identifying the root systems you're going to use.

My recommendation is VPN-based software. We use Open Text for our system silos [ph]. Also, you're going to be hosting, more than likely, in a virtual cloud. If you do do so, my recommendation is to go with a federal partner. It's a federal collab of hundreds of vendors out there, including some very well-known companies that have been vetted by the government for sanctioning or holding United States data in various levels of security.

Following that is resources, figuring out who it is that you're going to talk to and who is able to make the decisions that you're going to need to have made very quickly.

And lastly, hiring contractors or finding the resources necessary to actually do the building.

Lastly here, this entire concept and really the way the industry as a whole is going, is agile development. And for those of you who are not familiar with the term, agile system development implies that at any point in time we're effectively doing four different things. You are planning two months out so that you can identify what the priorities are and getting those stakeholders the heads-up that their request is going to be coming up, so they can start thinking through it.

You're going to be building a design document and holding job sessions with the stakeholders for the next month's development. You're going to be doing development in the current month effectively, and then you're going to be doing quality assurance and training and ultimately the release the last month there.

So it's -- you're basically doing 120 days' worth of work at any given point in time throughout the entire lifecycle of a system. But don't let that scare you. That just basically means that you have the ability to be flexible and move as the organization is changing quickly.

Very first month, so now we have a system, we have it posted somewhere. We've got the people and the resources necessary to actually get things started. So your first objective is going to be our grant intake. And that basically means you need to have -- reach out to your -- to the community, to your constituents, figure out who is interested in working with you in the

program. And we're going to be doing that by a Web-based application where you are going to receive attachments. And again, your SMEs, your subject matter experts are going to have to know what those are.

But more than anything else, you need to make sure that that foundation you're going to develop is flexible. No matter how good folks think they are, there are going to be changes to that application from the day that everybody says that they know what they want to the day you're actually done with the program. You're probably going to go through four or five, six variants of that application over time. So as you build that, make sure it's flexible.

Financial. Make sure that you have your integration with banks. However that paying process is going to work for you. It's going to be different for each individual or each organization. But make sure you have that connection.

And finally, accounts payable here. So the objective is all financial transactions associated with this grant should be filtering through your system. It is a way to hold accountability and ensure that all of the expenditures that ultimately, you're going to be doing transparency reporting on, ensures that it's all together in the same place.

And then finally reporting, you're going to need to figure out how much money you're spending. So a basic level of reporting at this point in time is pretty important.

Moving on to the second month here, again, we're going to continue working with our programs. The first item here is going to be starting the business workflows. So think about it from the perspective of, you now have applications. What do you do with the applications? Well, you need to review them. You need to screen them for eligibility, and you need to screen them for DOB, and ultimately your objective is to establish an award. There's obviously things that go along with that award. You're welcome to build that out as well.

Identifying grants status, so that you can start saying how many of those applications that you've reviewed and ultimately deemed to move on to the next step, or those that need to effectively be rejected. Staff roles are fairly self-explanatory, who's doing what, when.

And grant award calculations. As I discussed earlier, this can be very simple or it can be very complex. It depends on how your grant is structured. But again, this is another example of something that will be changing very frequently. And you need to make sure that is flexibly, not just for the future, but also to accommodate current awards.

And then finally, on this slide, we have financials. So we have a way of making payments now, we have a way of actually getting the cash out the door. Now we need activities. And you're going to set them up depending on your grant. In our case, we use DRGR as our conduit for HUD. You may use something else, but you're going to establish those activities both in that originating system, and those activities should be replicated here in the system.

Additionally, each of those activities should have budgets. So now you have said, here's the buckets that I'm going to be spending the money in, and here is the maximum amount of money that we are going to allow to be spent within each of those.

Over the next three months, at this point, and your organization may change your priorities at this point and may start to shift. And you'll see as I move on here, the timelines get a little looser. Within the next six months here, you're going to continue to build that business workflow case. Ultimately resulting in the ability to actually make payments to those homeowners.

Before we had talked about the ability to make payments out of the system. There's going to be a little different than what we're talking about here. These payments are going to now be specific to the grant. So you worked in the last -- up to the last three months, you've been establishing a grant award. Let's say it's $100,000 for a specific homeowner that you're going to do some repair work on their home. You need to be able to establish that as an obligation and then to make individual payments against that obligation.

And it's not appropriate to say, well, I have an activity code, which is my ceiling. Your activity code for the thousands of applications that you're probably going to get is hundreds of millions of dollars. So you need to be able to establish that payment structure and that obligation structure within each of the individual applicants' folders.

Accounts payable then back to that same thing. And finally, program reporting need to be established as well.

On the financial end, you need to be able to establish effectively the hierarchy of your grant management. So it starts, at least for us, within grants. So again, flexibility. My recommendation is to build from more than one grant, but have the hierarchy start at the grant level, followed by -- and some cases trump this, but in our case, programs, followed by rounds. If you have rounds, not all grantees have rounds, so if you don't, don't worry about it. But if you do, you get your rounds.

And then you finally have your individual activities. And again, your activities now slide within those programs. And followed by that, we have QPRs. At this point in time, you're going to be expected to start reforming all QPRs. Our system has all the expenditures which you can pull them out and adequately associated with the individual activities.

Okay, first year wrap up here. And then, everybody's going to do a little different. Effectively, what you're going to be doing is cleaning up the program. You're going to be finishing with workflows. It's a good opportunity for you to now go back to all the stakeholders and the managers that are doing individual work.

Have conversations with them. You may have assumed that a single workflow stage encompassed a certain type of activity, but once they actually started doing it, they realize that their application within that individual activity or workflow step. It's a good opportunity for you now to take that single workflow step and break it into several different workflow steps, if necessary, adding validations and restrictions.

Again, if somebody has got to fill something in, or they're supposed to fill something in or add an attachment, make sure that's actually getting done before they're able to move on in the work.

Well, grant revisions. I can't tell you how many times I've heard that a grant won't change or can't change, it shouldn't change. And then just a handful of days after the program is actually approved, the first change is there and ready. So from a systems perspective, be prepared to deal with it, have a plan to deal with it. Whether you actually build it proactively or not, it's going to be a decision for you and your management team.

And reporting, again, will focus on data cleanup, make sure that all the fields are filled out correctly, dashboard transparency, and finishing effectively the QPR metric end of reporting. Anticipate technical monitoring and systems and being able to record that in your system.

And then within financials, you may choose to refine even further your financial expenditures. In our case, we have things called object codes and general ledgers, GLs. Your organization and your payment structure may be a little different.

And then finally, forecasting, as we all know, any time that you have an actual moment, a revised forecast is necessary to be set out.

Now we really get into the fun. So now we're out of the fire. We are not necessarily every single day having five people come to us saying that I can't do my job unless this is happening. This is now the opportunity you have to look back at the system and to really start building efficiencies into your organization and integrating them into the system.

So this is integrated financial systems. What is that? We call it IFS. Effectively, this is a way to organize your finance department, to put it in in clean terms. We have, in our specific grant, we are managing three grants, over seven rounds, over 35 programs and over 1,000 activities. That's really light. It's probably closer about 1,700 activities.

To say the least, everything needs to tick in time all the time. And you need a system to effectively ensure that when you are taking money away from one activity, that you're putting it back in another activity. That when you are making those movements, those same movements are occurring -- in our case, we have three systems. So we have the state of New Jersey's accounting system. We have our Syrons system. That's our grant management system. Now we have DRGR.

So this is definitely important for you to ensure all those things maintain consistency and are coherent and talking to each other. You can decide to build a workflow into that, critical metrics associated with the activity. So that's another thing -- more often than not, especially in the beginning, you're out there, you're trying to create activities so you can segregate your financial elements.

However, as time progresses, you realize, hey, our QPR reporting needs to actually encompass elements of metrics that are beyond financial, making sure that every time you create a new activity, have the metrics associated with that.

And then lastly, and this is way down the road for most of you, it's closeout. That as you think about closeout, as you think about all the individual groups that are involved in your closeout, closing out individual activities, programs where you've got 30-plus programs, there's a lot of people that need to put eyes on that because you just go and close an activity or a program.

This is an example of IFS. Feel free to take a closer look at this as you have time.

QPR management. I won't spend too much time on this, but effectively, this is saying over the course of our grant, we have processed over 10,000 QPR -- we call them folders. Effectively, it's a combination of an activity and a QPR cycle report. Where the property goes into that.

When you start with over a thousand activities, you have to figure out what activities you're going to be reporting on during this quarter. Why am I reporting on them? What kind of information do I need to supply in the QPR cycle? We built a system that actually pulls information from helpful sources throughout the system and actually helps the user, helps our finance team decide what filters need to be selected and gives them the ability to overwrite that system recommendation.

Next slide here is compliance and contract management, and effectively, this is to say you as a grantee are going to have lots of contracts. Those are probably very expensive contracts. And there's probably a high chance that they get audited at some point. Having a single point where all of the documentation associated with that contract exists, and that, similar to the hierarchy we just talked about on the financial and the things, this is another hierarchy that needs to exist.

You have your contract ceiling, which was part of your RFP, or procurement process. You then have individual tax waivers, potentially, based on the segmentation of data. You then have purchase orders, more than likely, which is your elements of even [inaudible] within your purchase orders. And then you have individual invoices.

In addition to, depending on your contract, 30, 40, 50 lines within that contract. And every one of those layers has their own financial segmentation that need to be managed within. So having a single place to pull all that together and making sure that it's all ticking and tying, and audit-proof or at least auditable, is very important.

And finally, here on bottom compliance. Basically ensure that your system is tracking what people are doing in the system. Every single time somebody touches something in our systems, [inaudible]. There are millions of audit [inaudible] that represent [inaudible]. We're probably well above that at this point, but effectively, just the ability of figuring out when something changed, and you don't understand why, being able to figure out what was changed. And I think that does it for the deck, here. Olivia?

Olivia Healey: Thanks, James. So we have some good questions coming in through the box right now. And just as a reminder to add any as you think of them. For our first question, I'll start you off with, I wanted to run this one by you. So what processes do you have in place to manage costs while accounting for the policy changes throughout the program?

James Shuster: Great. So the first one, first response to that, I will say, is a single point of approval. When I started in the organization, we had a methodology where practically any manager could go and request a change. Quickly gets out of control if you allow that to happen, to say the least.

So you really need a centralized point of management that will help you set that financial structure within your consulting role or your systems development. And also give you the opportunity to centralize the prioritization associated with those changes. Additionally, having a clear understanding of the effort a specific thing is going to cost. Some of this comes with experience. Some of it comes with building the relationship with the group that's going to ultimately do the work.

But you will have -- to changes that sound at face value relatively similar, one of which could cost $100,000, the one may cost $10,000. Because of the backend structure that needs to be done, you may have to bring in a new server to accommodate a certain type of integration. So understanding what that cost is going to be.

And finally, as I noted earlier, the ability to build flexibly, ensure that as you have the time and the resources to do some [inaudible], building it flexibly today will save you a ton of money down the road.

Scott Ledford: Great, thanks, James. Next question for you. Do you have any systems procurement lessons learned? Like what should go in the scope of work? How should the payment be structured, et cetera.

James Shuster: Great question. So I will say that it is important to not be overly specific in your scope of work. Especially in the beginning of your grant, you don't fully understand what it is that you need to be asking for. And, yes, this is what you're effectively asking here. But the truth of the matter is, every grantee is going to be different. Everybody's names are going to be different.

So trying to be reasonably generic while still being able to create a structure of maintenance and development that you can build around and control. Structuring the contract, especially in the beginning, time and material-based methodology is going to also be very, very critical.

I'm not really a general proponent of time and material contracts. I'd much rather milestone-based contracts. But at the beginning, again, you just don't know what it is you're going to be asking for. So both from your end and the contractors end, it's difficult for either of you to accurately [inaudible] the ask.

It's a lot easier to amend or adjust a contract for additional dollars than it is to change the scope of work on a procured contract. And also, I would very highly recommend liquidated damages. If you don't have them, they're a great tool. It's effectively your stay. If you build your liquidated damages correctly and you take the time to think about what kinds of things that you need your contractor to do, the timelines you need to do them in, your liquidated damages is a way for you to control the activities of your contractor without having to jump to the very large step of saying that the contractor is breaching the contract. So it's a good moral ground.

And finally, for folks -- you're welcome to look at the contract that we have for silos. It's available publicly on New Jersey Start if you start gen work open contracts, you can look for Syroms. You'll see what was out there as our request for proposal and the way it was structured. So resources available to us as the public.

Olivia Healey: To kind of build on that a little bit more, why did you decide to build Syroms to account for multiple grants?

James Shuster: So effectively, the answer in my case is relatively simple. It was, we had an additional grant that was a small piece of the larger grant that we were doing. In our case, it was NDR. The structure of the grant was almost identical to our final CDGB-DR grant. And so, as such, and the way it needed to be managed, it allowed us to build that hierarchy that we were talking about earlier, and effectively just adding another layer onto that hierarchy so that it could build out its own structure.

And then also, it depends on the cost of your system. Ultimately, we took a lot of time and effort to build a comprehensive end-to-end a system that's going to accommodate all aspects of grant management. So it's important that we take that value and effectively amortize it over the course of the next 20, 30, 40 years as we have future disasters, rather than just looking at it as a single point in time expense, and then do it again next time.

Scott Ledford: How do you ensure you get a system that is right sized for your needs and not paying for features that you do not need?

James Shuster: Decide, basically on how to integrate their systems and your staffs. So understand for your specific organization, where does that one lie? In our organization, systems are effectively an SOP for staff. If you know nothing about our program and nothing about what you're supposed to do, our systems are guide rails by which you cannot proceed through the grant process or through a payment process without ensuring that checks and balances have been met before you proceed.

Other grantees may decide that they want an internal process and the flexibility of having a less validated system. Really, it's going to be up to you and your organization. Additionally, I would say majority is key. As you build components into your system by building them in a larger [inaudible], that allows you to have pieces that you can plug in and ultimately remove when they're no longer necessary and archive.

So it allows you, as your grant life cycle continues to post down the size of that system, the servers that are required to support it, and ultimately still be left with that core matrix of the system, which has in perpetuity the ability to support your grant and future grants down the road.

Olivia Healey: Great. Has there been any DR-specific features you found yourself needing that the traditional accounting systems didn't have? For example, modules or, that could be an example. Doesn't have to be the exact example.

James Shuster: So DR-specific would definitely be QPR. The ability to identify your expenditures, identify where you've met natural objectives and reporting out on them. I would really say that the most unique element of disaster recovery versus all other grant management systems that I've built is just, very frankly, the time. You have an absurdly compressed timeline. And there are people just constantly, not necessarily on your back, but across your organization's back in public, in news broadcasts about how money isn't being spent yada, yada, yada, effectively, that partially lands on you as a system designer and developer to make sure that your systems are not standing in the way of allowing that to happen.

Scott Ledford: Great. We got some great questions here, so folks, just a reminder to keep sending them in through the Q&A here. Let's see, how does your information system interact with your accounting system? And how do DR staff handle this?

James Shuster: So our accounting system is our DR system, effectively. So if you were to look in our Syron system, you can see every financial transaction. Going, coming, budgeting, obligations, forecasts, and everything is within our -- within our grant management system.

More specifically to the accountants in the room here, we have a back-end integration with our -- with several banks, actually, where when a request for a payment is generated, we generate a file and we send it through basically FTP process to the bank. The bank generates payment, sends a check out. We then have another EFT process that actually reads files directly from the bank. We then load that information back into the system so that it's not just accounting from the fact that we requested the payment, but it's also accounting for the status of those payments, check numbers, ACH numbers, and also the possibility that the financial transaction, for one reason or another, didn't go through. We need to cancel, resubmit it, whatever the case is. So that's probably where we are.

Olivia Healey: Great. Any stories about your decisions or tradeoffs or lessons that you've learned, that would be a great example for other DR grantees who are wrestling with the same things?

James Shuster: Sure. So that's probably going to be depending upon the way [inaudible] in your organization. I'm going to take a little part in what we built over the last two years here, which is we build the TATM. Basically it is a modular program platform within our Syron system.

And effectively, what that is, is it's an end-to-end grant management module that is blank. You can take it and you can copy it and you can put it into production within an hour. And within that module, you basically have the ability of having an online application. The initial stages of

workflow associated with the application review DOB. It has a comprehensive calculator that has the ability of applicant interactions, organizing, and maintaining attachments into all case loads and funding.

And all of that is ready to go day one within an hour. We can have a program up and running. And then from that date, because the entire thing is being built on a back -- on the back end .net framework, anything can change within that entire system. Again, within a matter of minutes. We can we can create 15 fields that automatically show up on our data reports. We can adjust the applications on the fly.

So if you have the bandwidth, I highly recommend taking the time and thinking through the best elements of everything you built, put them all together so that you can take that, copy it, and apply it for future programs down the road in a very quick and modular capacity. I would say that ever since we've done that, we've already made four copies of that modulation within the last year alone. So it's become very helpful and it saved us a lot of money.

Sam Vavintine: And if I could just add to that as well, from an overall management perspective and not from the IT perspective or technology perspective, I think the system that we have has done a couple of things that I think are very important for us. One being, you know, we've beaten back audits from the OIG. We've held up under tough scrutiny from CPD monitoring, our own internal state licensure for auditing.

The system has been really the big -- a big component of the success that we've had in fending off those auditors. And I think, you know, if -- without the solution, I think there would have been some significant findings that the state may have faced because of our inability to show information in a comprehensive way, whether it's to the OIG or any other agency.

The other thing that I think is important that, you know, the life cycle of DR, you know, for some grantees is not permanent. I know that there's, you know, jurisdictions around the country that, you know, just are perpetually in this life cycle. But for others, and maybe New Jersey is in that in that group where, you know, we don't get struck with a Sandy-type disaster every three years.

And so, you know, the staff needs to right-size every so often. And that means, you know, attrition, that means reductions in staff, personnel. But what hasn't changed and remained a constant is the system that has held together the business of our division and our operations through the life cycle of the grant. And, you know, as James has mentioned, you know, even gotten stronger as we've gone on through this process.

Scott Ledford: Thanks for the [inaudible], Sam. So the next question -- probably hit on this a little bit, but just if there's anything you want to clarify in here. What program and financial metrics do you have in place, and what software tools are utilized for this process?

James Shuster: I would probably refer you back to some of the slides that we have here. The one that comes to mind the most is our integrated financial management, or IFS system, which pulls all those elements together into a single uniform data point.

Olivia Healey: Sorry, I was trying to see if I could find that slide for you to pull up without going through them all.

James Shuster: Oh yeah, I can pull it back up if you'd like to.

Olivia Healey: I think it's -- I have -- it's here, slide 16, right?

James Shuster: Sure. Yes. That would be great. So I can -- if we've got the time here, I can try to explain what it is that we're looking at. So effectively, this is the hierarchy that we're talking about here. So you've got grants, rounds, programs, and then down here below that would be the individual activities. And in our case, you can see -- and this isn't a comprehensive view, this would be an element of a change associated with this.

So it's a little difficult to see the full picture in this particular screenshot. But you can see here you've got your original allocation, what the current allocation is, beginning and end dates of each individual element, and that carries over and down. Again, I don't want to get caught up in rounds. I heard that we are potentially limited in rounds, or have a limited rounds. It was a fun time for us. So we won't get into that too much.

But that same structure arches over a down throughout the remainder of this methodology. But ultimately, when you make an adjustment, either a positive or negative adjustment, the system requires a counterbalance with other elements of the system. And only if you were to go up to the very highest level, and I am going to reduce the grant as a whole, would it allow you to make a -- an offsetting adjustment that wasn't counteracted somewhere else. So again, this brings it all together and make sure that everybody financially is talking the same language throughout all of your systems.

Olivia Healey: Great, thanks for that extra detail. We have another question that came through on what basis do you recommend BPM software, and what is your selection criteria?

James Shuster: So the basis is fairly straightforward. It's -- our organization at the very least, considers the incremental steps that a individual application needs to take throughout the course of its lifecycle. So it starts with a pre-application and the application and multiple layers of review, eligibility, DOB, yada, yada, yada.

In our primary program called WREN [ph], we have over 100 individual workflow stages associated with a single application. BPM is just, quite frankly, designed to be able to take that application and be able to hand it off to individuals throughout that workflow. It also allows you -- and we say, well, we could create a stage in any system to be able to do that. The problem is, is being able to develop those layers within that workflow step that basically said, hey, we're in construction, but, while in construction, we're doing an amend.

But while we're doing an amendment, we're also updating another data point. So there's potentially three different layers at play on the workflow. And depending on the reporting requirements, you still need to represent that application as a construction application, even

though technically from a stage perspective, it's somewhere else. So BPM, which is a category of software, is the right tool for that type of a job.

There are other tools that may be better suited for your organization, including accounting-based systems. Ultimately, we took our BPM system. We loaded that net on top of our open text .net system and we found that we effectively built an accounting system on top of BPM, which allow everybody to have a role in every type of module and assign those rules and limit access.

And then ultimately the criteria associated with VPN. To be honest with you, I haven't been out there shopping for VPN software in the last five years. So there are probably better people than me to be able to speak to what the right software is out there today. I believe the industry as a whole has come a long way in the last five years. But I would, again, look through the slides that we've got here, think through how your program is going to integrate with the system and ultimately make sure that your system that you've selected is able to accommodate your specifics.

Scott Ledford: Great. Here's a fun one. Does all data need to be encrypted or just certain data?

James Shuster: So great question. I will start by saying I am not a security expert. I am a systems designer. I am, unfortunately, a fair number of other things. And I do know about security, but we have people that are better suited to really answer that question that I rely on.

That being said, at a 10,000-foot-view perspective, the system as a whole needs to be encrypted. Forget about the individual elements of the system. The entire system as a whole needs to be encrypted. It needs to be behind the firewall, it needs security measures, protocols in place.

Again, I referenced earlier [inaudible]. Look into that. It's a federalized standard full system security and hosting security, as well as your system as a whole is behind firewalls and behind those elemental layers of protection. Everything is behind that firewall is also secure. Also, I'll add, don't have excess PLI in your system if you don't need to.

In our case, we don't have social security numbers, necessarily in our system. So we don't have by the Department of Defense documentation in our system. So the only encryption that's necessary really is going to depend on your specific organization, the types of documents that you are storing in your system. And I would rely on your organizational experts or system security to really be able to answer that question correctly for you.

Olivia Healey: This could be a hard one for you, but it's always a good one from the attendees' perspective. But what's one piece of advice that you could give someone who's new to CDBG-DR, and is overwhelmed with the system infrastructure needs?

James Shuster: Infrastructure needs, specifically. I'm going back on my statement previous, which is I'm a designer of software and not necessarily the infrastructure. There are a lot of infrastructure needs, no doubt. I think at one point, we had 64 servers across four different elements -- we had a production system, a test system, a pre-test system, and a development system.

And then we had that redundant in two different data centers. So it's no joke. And I wish I could give you more advice other than to just say, lean on your experts associated with infrastructure, and when your grant starts slowing down, don't be afraid to start cutting back some of those redundancies.

Over the last year, we've reduced our DR capacity, since we're a DR organization -- we reduced our DR capacity such that we have tapes still. We have an external -- basically, we're using BMF. We have external copies of our BM servers and we're also replicating our database, our whole database offsite.

So because we're kind of ending the tail end of our grant management cycle, the need for a disaster to occur and within two hours to be up and operational again really isn't as much of a concern for us. So we've decided to save significant money in our hosting cost to accept a 24-hour return to a production scenario. So don't be afraid to make those decisions. Sometimes they're hard decisions to make. But again, your objective is to keep costs low and proportionate to the level of activity needed in your system.

Scott Ledford: Great. Thanks, James. So I've actually come up with a few brief questions myself that might be fun to do a little run with both you and Sam to get your read on these. I'm going to call this inside the grantees' office in homage to James Lipton Inside the Actor's Studio. If you guys have ever seen that. But so, some fun questions for.

Staying within the context of disaster recovery and mitigation, of course, what is your favorite word, phrase, or concept?

Sam Vavintine: Well. I'll go first, James, I'll go first. So what I think is a metaphor that I hadn't heard, certainly without the frequency that I heard it before I moved into the DR world was -- and specifically as it relates to the system, was, we're building the airplane while we're flying it. Right. And, you know, it's a scary concept. And after you hear it for the hundredth time, it's annoying. And it's just not fun to hear because it is what it is.

I mean, you don't have both wings and you're up in the air and you've got all of these different forces and they call it turbulence, just pulling you in all sorts of different directions. But yet, you know, the system was nimble enough for us and what we did across DR to really respond to those, you know, different policies, different political forces. It was -- looking back on it, it's actually a very fitting description.

James Shuster: I think Sam's going to get the Gold Star for his phrase. I wish I would have thought about that, but that's absolutely the best phrase that we could possibly use, and it's absolutely true.

What came to mind for me is return on investment. Basically trying to say, are we going to spend the dollar today and then get a dollar for the goods, or are we going to spend $2.00 today and over the course of the next 10 years, getting $100 for the goods? Just trying to kind of keep that proportionality in mind with every decision that you're making and figuring out, is it worth it to

spend that money in the system today or are there more cost-effective alternatives to spending that dollar?

Scott Ledford: All right, great answers. And you may have answered this one already. Is there a word or a concept you would be perfectly happy to never hear again?

Sam Vavintine: I wasn't ready. I don't know if James has any.

James Shuster: What comes to mind is probably, we don't have the money to do that. And it really pertains less to our DR scenario right now and more to my history as a project manager. And actually, to a degree, and maybe apply to some of our grantees out there as well that are managing smaller grants and not billions of dollars.

As Sam has illustrated with this building a plane, quite frankly, while it's flying, scenario, about trying to build that plane while also being told that you only have enough money for two bolts is quite a difficult feat. And, you know, I'm a fiscally minded individual, but at some point, you need to spend some money to get what you need out of a system. So having the budget to do what you need to do is something that I'll never hope to hear again [ph].

Scott Ledford: Sam, anything you want to add on that?

Sam Vavintine: Well, I mean, we're talking about, you know, not relative to any systems facet of recovery or DR. I mean, there's lots of things. I never want to hear OIG again. I never want to hear CPD monitoring, you know. Although I do love my friends at HUD, but, you know, those meetings are not always, you know, enjoyable, outside of the personal interactions.

But like I said, I mean, the system eased all of that pain on us. So you know, that's probably the most painful part of everything that we've gone through, is just trying to run a recovery as well as we can while also trying to respond to people's questions and audits. And so that facet of the recovery is not something that I enjoyed nor ever want to think about again. But who knows, maybe I'll have another audit coming my way. Right?

James Shuster: So I actually just thought of another one to tie in with what Sam was just saying here. And that's a phrase, it's a double-edged sword. As you think about building systems and are building the systems, creating that transparency, it can be both a blessing and a curse. If you are not managing your organization well, if you don't have the validations and checks and balances in place through your own policies, SOP, so on and so forth, and you're building a comprehensive system that allows everything to be seen, it's going to be there for the auditors to find.

So it's not to discourage anybody from building a comprehensive system. It's just, make sure that when you're doing that, don't -- [inaudible] lean on your systems, you also have to continue managing the process and the people throughout that.

Scott Ledford: Great. All right, well, Olivia, I probably consumed enough time on those. So we want to get to the resources.

Olivia Healey: Yes, we do. So for our resources, we have our typical DR and MIT pages located on the HUD exchange. And then throughout this entire presentation, we have so many great, wonderful examples that you guys can reference throughout.

We've provided contact information for Sam and James. So thank you guys so much for putting your contact on there for others to reach you and provide additional questions or follow up.

Thank you guys so much for attending, we made it to the end of our day 2. We've learned so much. I know that it's a lot of information all at one time. So keep up the good work of listening and having these details. And you can always reference this content on the HUD exchange after we get it posted. And we look forward to seeing you all tomorrow.

Sam and James, thank you so much for your time. You guys did an incredible job and clearly outlined a lot of your best practices and really applicable skills that everybody can use. So thank you.

James Shuster: And thank you both again.

Olivia Healey: Perfect. And so, you guys can log off, and have a great day.

(END)