Cyberseminar Transcript

Date: April 11, 2019

Series: VA Informatics and Computing Infrastructure

Session: SAS Grid-Enterprise Power Above and Beyond Open-Source Technology

Presenter: Mark Ezzo, BS

*This is an unedited transcript of this session. As such, it may contain omissions or errors due to sound quality or misinterpretation. For clarification or verification of any points in the transcript, please refer to the audio version posted at* [http://www.hsrd.research.va.gov/cyberseminars/catalog-archive.cfm](file:///C:\Users\VHASLCMyersK\AppData\Local\Microsoft\Windows\Temporary%20Internet%20Files\Content.IE5\73RXHSNF\l)

Robert: And as it's just at the top of the hour now I'd like to introduce our presenter, Mark Ezzo, who is a Data Scientist and a SAS Administrator at VINCI. Mark can I turn over things to you?

Mark Ezzo: Yes Sir. How are we looking?

Robert: Looks good.

Mark Ezzo: Okay. Good morning or afternoon folks. As always it's a joy to present to you and today we're going to take a bit of a different approach. I'm not just going to show you a lot of technical things. We're going to talk a little bit of philosophy, culture, and technology. So today it’s about SAS Grid Above and Beyond Open Source. Now saying that does not mean that SAS is better than open source or open source is better than SAS. We're going to be looking at the difference between Enterprise technology and open source and how the meld of them is actually, probably going to be our best option. So let's proceed.

And we're going to start off with a poll question, and the poll question is, what is your primary usage of the SAS grid? A. ETL, B. Data Preparation (SQL & SAS), C. Analysis (Including Modeling), D. Other. And we’ll give you folks a few moments.

Robert: Yeah, I have about 50% who have made their choice already, so we'll give people a few more moments to make their choices. And it looks like things have leveled off at about 75%. So I'm going to go ahead and close the poll, Mark and share out the results. And only 3% said ETL, 41% said Data Prep (SQL and SAS), 45% chose Analysis including Modeling, and 10% said Other. So I\_\_

Mark Ezzo: Very good and that's quite anticipated. Well done. Now to proceed, let us now go to poll question number two. What is your primary usage of open source? Now that would include R, Python, Java, whatever you desire.

Robert: And that poll is open and people are making their choices. Should we warn our attendees, Mark that they really need to pay attention right now because there's like 10 polls coming up?

Mark Ezzo: Well maybe not quite that many, but this is going to be a more interactive session, yes.

Robert: Yeah. All right we have about 60% voted. And things usually level off around 75 or 80, so again just a few more moments. I'll go ahead and close the poll and share out the results. For some reason my screen is displaying what's remained from the last poll. So I'm having a little bit of technical difficulties, here. Let me try that again.

Mark Ezzo: That's okay, technical difficulties keep us all employed.

Robert: Right. Okay it's better now. The question was what's your primary usage of open source, right?

Mark Ezzo: That's correct

Robert: Yeah, so 0% chose ETL, 28% chose Data Prep., 62% said Analysis, and 10% Other.

Mark Ezzo: Oh very good that's all as anticipated.\_\_

Robert: Great.

Mark Ezzo: Okay, to proceed. Now let's talk about Enterprise versus open source. Now I've gotten a lot of, essentially call this a lot of things off of the web and a lot of discussions and all. And I see a lot of blogs talking about, technology A is better than technology B, etc., etc. But I think that is kind of becoming a bit passé, I think what people are looking at more is what tool works best for them. So let's look at and examine it. Open source software refers to the software which uses the code freely available on the Internet.

Robert: Mark, I'm sorry it wasn't displaying correctly again I just had to send you the pop up. I'm sorry\_\_

Mark Ezzo: Oh, okay.

Robert: Okay good.

Mark Ezzo: Good, okay. Again open source software refers to the software which uses the code freely available on the Internet. The code can be copied, modified or deleted by other users and organizations. As the software is open to the public, the result is that it constantly updates, improves and expands as more people can work on its improvement. In a word it's what, I like about open source I think what many people do, if there's a new statistical say methodology or technique out there it can come up pretty quickly via open source. Whereas on Enterprise product like SAS or SPSS or SAP, etc., though essentially they might release it quickly but never as quickly as open source, in my opinion. It may be more reliable but we’ll discuss that later. Closed source software is opposite to OSS and means the software which uses the proprietary and closely guarded code. Only the original authors of software can access, copy, and alter that software. In case with closed source software, you are not purchasing the software, but only pay to use it. As we do with, as an example, SAS or SPSS, etc., we are essentially, even though we do own the code, we have to pay a yearly fee for updates, for technical support, etc. So one is a little bit wide open, the other bit is, in my opinion, a little bit too closed but there's a definite difference there. Okay, now why am I not progressing.

There we go. Let's make sure everything, good. Here is a comparison of five basic aspects, pricing, security, support, source availability, and usability. Out there that's pretty much what most of the discussions are around. Let's talk about price policy. Obviously this is where open source is going to be a big winner. Though not as big a winner as many people think, because, for example Linux is an open source software. However you will pay a hefty fee to get support from Red Hat. Open source often referred as a free of cost software. It can, however, have costs for extras like assistance, additional services or added functionality. Thus, you may still pay for a service with OSS. And that is very much true, we have R Studio, you may remember Revo-R that Microsoft purchased. Closed source software is usually a paid software. The costs can vary depending on the complexity of the software. SAS of course is very complex and has been around since the 70s and has evolved thusly. While the price can be higher, what you get is a better product, full support, functionality and innovation. Now I know some people take slight umbrage with that. However, most companies provide free trials to convince the purchaser that their software is the right fit. Now that is very true.

Security, now this is the trickier one. The question of security is very controversial as each software has two sides of the coin. The code of open source software can be viewed, shared and modified by the community, which means anyone can fix, upgrade and test the broken code. The bugs are fixed quickly, that's usually true though I've seen instances where the bugs just proliferate, and the code is checked thoroughly after each release. However, because of availability, the source code is open for hackers to practice on. That is really one of the main reasons why private industry doesn't embrace open source as much as academia or the government does. They are extremely concerned about hacking. However it's making its inroads but I can tell you from experience a lot of times it's very much walled off. On the contrary, closed source software can be fixed only by a vendor, this is rare. If something goes wrong with the software, you send a request and wait for the answer from the support team. Solving the problem can take much longer than compared to open source. And that is correct, though I've seen Linux get bug fixes pretty quickly, SAS will send out a hot fix as soon as it can, but again it's not quite as fast as open source. When it comes to choosing the most secure software, the answer is that each of them has its pros and cons. Thus, it is often a challenge for firms which work in a particular industry.

Now let's talk about quality of support. Now this is where I think regardless of blogs, I think this is really where Enterprise software has a big advantage. Comparing open source and closed source software support. It is obvious that CSS is predominant in this case. The costs for it include an option contact support, and get it in one business day in most cases. That's true with SAS, SAS has a 24 hour maximum turn around. Usually if we send something in to tech support we get it before the day is out. The response is well organized and documented. And I would agree with that. For open source software such option is not provided. The only support options are forums, useful articles and hired expert. However, it is not surprising that using such kind of service you will not receive a high level of response. That's true, and I'll say this, that a lot of forums and blogs out there that I found to be very informative and very useful. However if it's a question that is quite unique, something that’s popped up, it usually will take time for someone to respond where Enterprise software can respond much more quickly. Now on the other side of the coin, source code availability. Now obviously open source is going to win this one. Open source software provides an ability to change the source code without any restrictions. Individual users can develop what they want and get benefits from innovation developed by others within the user community. As the source code is easily accessible, it often enables the software developers to improve the already existing programs. That is a very true statement. With the source code being available you can look at it, you can learn from it, you can update accordingly, send it back out, and once it's validated out there it becomes part of mainstream. Closed source software is more restricted than open source software because the source code cannot be changed or viewed. Obviously this is how they make their money. However, such limitation is what may contribute to closed source software security and reliability.

Now usability. Usability in this case, well let's read it. Usability is a painful subject of open source software. User guides are written for developers rather than to layperson users. I don't know how true that is anymore, I think it's gotten a lot better as far as for laypersons. Also, these manuals are failing to conform to the standards and structures. That's probably a little more true. For closed source software usability is one of the merits. Documentation is usually well written, contains detailed instructions. The difference is, you're getting the developer or users writing the software as opposed to a professional technical writer. And there, closed source will have a bit of an advantage.

Now, here's something on Enterprise advantages, and again that's nothing to say it's better it's just a commentary shall we say. Companies can meet their engineering needs within minutes, and Development teams never need to follow convoluted procurement cycles. Now that I will take umbrage with, anyone who's had to make a purchase in the VA knows that procurement cycles can be quite convoluted at times. Enterprise software improves efficiency and offers strong security features that many industries consider essential to their operations. That is very true. Flexible licensing systems, pardon me please, combine multiple licensing arrangements under single umbrella contracts. That is true in many cases, not in all cases of Enterprise software. It's a little more true in SAS where you can have a license, site license, you can have product's accordingly, sometimes though, with like what we had issues with SPSS and IBM, and it was not so easy to make any changes. Finding, testing, and verifying and hardening open source software entails many unanticipated costs for maintenance, internal development and generating user-friendly support processes. Enterprise software solutions eliminate these worries and expenses, and you have a support team ready to take your call and provide answers or fixes when needed. That is probably a little more true. And here's one that you may or may not be aware of. To qualify as truly Enterprise-grade, databases need to ensure, what has and is still called, ACID transactions. The ACID acronym refers to atomicity, consistency, isolation and durability, and transaction processing needs to satisfy its guidelines. Atomicity, data changes need to function as single transactions, causing simultaneous updates in related databases. Consistency, zero-sum totals must balance after inputting data updates. Meaning we don't have errors. Isolation, transaction processes remain invisible until posted simultaneously in all related databases. Meaning you're not just viewing, like an update table, which many databases do, it’ll have a table out there that's temporary that's available until they writ, the production databases update its SQL server. Of course that's the same thing. Durability, changes in databases persist even after catastrophic system failures. Meaning if we have, let's say we have a site that burns down when the backups come back the database appears the same to you. There's nothing that has to be re-created by developers etc., etc.

Now another poll question. Which venue, and there's going to be about three or four of these coming right at you, which venue is your preference? A. Enterprise closed source, B. Open- Source, C. Combination, D. No Preference, E. No Opinion.

Robert: Okay, people are making their choices. Fairly quickly this time.

Mark Ezzo: Now this is something where people have an opinion.

Robert: Alright things have leveled off, just around 70%, so I'm going to go ahead and close the poll in a second.

Mark Ezzo: Yeah.

Robert: And share out the results. And Mark, 47% say it's a Combination, and well I'll just go through them, 13% said Enterprise, 17% Open-Source, 47 Combination, 6 No Preference, and 17% No Opinion. Shall we move on to the next poll?

Mark Ezzo: Yes Sir, and I would agree with that wholeheartedly. That combination is the way to go.

Robert: Okay.

Mark Ezzo: Next one, if you chose Enterprise Close-Source, please pick the primary reason?

Robert: And that poll is up.

Mark Ezzo: Pricing, Security, Support, Source Code Availability, Usability Documentation other. And both of these, next polls are obviously going to have a smaller sample size.

Robert: Right.

Mark Ezzo: I think we have a well-informed group here they are responding very well. I would highly agree with combination.

Robert: Seems like this one is taking a little bit longer, people are thinking about their answer choices here. Some things have leveled off, haven't changed in a few seconds so I'm going to go ahead and close this poll and share out the results. And what we have is 0% said Pricing, Security got 17%, Support was the largest with 43%, Usability next at 30% and Source Code only had 9%.

Mark Ezzo: Great.

Robert: So support and usability were the big ones.

Mark Ezzo: Yep, I would expect that. And now if you chose open source please pick the primary reason, same question. Pricing, Security, Support, Source Code Availability, Usability Documentation other. And while this may seem tedious to some folks we have a very good reason for doing this.

Robert: So everybody who chose open source is now making their choices. It's starting to slow down I can see. It's leveled off at about the same rate as the last question so I'm going to go ahead and close this poll and share out the results. And Mark, Pricing and Source Code got 35%, Security 9%, Support 4%, and Usability Documentation 17%. So pricing and source code availability were the big ones.

Mark Ezzo: Yep, that would be as expected. Okay folks let's proceed. To our last one and I apologize if you find this tedious. If you chose Combination, please pick the primary reason. Pricing, Security, Support, Source Code Availability, Usability or Other. Now much larger sample size in comparison.

Robert: And that poll is up, and answers are streaming in. It started to slow down. Yeah, and it's about the same rate as the last two questions, so I'll give people another second or two. And I'll go ahead and close that pole and share out the results. And I'll tell you, Mark that, 29%, I'm sorry Usability at 42% was the highest, and Source Code Availability at 29% was next, then Support at 13%, Pricing at 13% and Security at 4%.

Mark Ezzo: Sure.

Robert: And I believe that is it.

Mark Ezzo: I believe that is it. So now that we've had our American Idol moment let us move on.

Robert: I'm going to have to do the trick that we did before though\_\_

Mark Ezzo: Please do.

Robert: Sorry about that, I don't know what's going on but I have to take it back, show my slide, and then send it back to you. Here you go.

Mark Ezzo: There we are. Am I visible?

Robert: Yes you are. Thank you.

Mark Ezzo: Okay, good. Now, as far as public code repositories. Both SAS and open source have these code repositories, and I just picked a few, there's going to be more later, for about anything that you can desire, here are some examples. SAS has a very large one on GitHub. R, that's I find it to be a nice one, R Forge, that's r project dot org. Python, Python tips yadda, yadda, yadda. Python libraries you can't live without. Etc. Like I said SAS has been around, since the 70s, and there is, if you know how to search on the Internet, you can find anything your heart desires about SAS on any product forums, or rather, and very enthusiastic folks. And I'll say even though R and Python are much newer, I'm very pleased with what is out there on the Internet also. So essentially the new generation, and even we in the older generation, have quite a bit of resources at hand, all at our fingertips.

And this is, now let's talk about our current grid and how we are combining things and what our future may be. This is our current SAS grid. Capability, centralized shared workload management. Why does it matter? Effectively manage jobs and users, prioritizations, compliance and auditing. And that is all very true, as people, many people know, everything by, in SAS is all on the grid is authenticated through metadata. We create your profile's, we handle security, we do not abrogate the security for example of active directory, for folders, or the security of databases. But we do control what security you have within SAS, and what you can get to. We have high availability. Now high availability means if, we for example, we have an operations grid which is one and both grids are the same. One is a metadata node, there are three compute nodes, there is a gsub node, which is that's processing and a mid-tier node upon which we have Enterprise Miner and tools that the SAS Admins use. And what it means, if we lose for example, we have A, B, and C compute nodes we lose any one of those, we're still in the game. B and C will work if we lose two, C will work. Now granted, things may slow down slightly but you'll still be able to process. There’s many times we will take out a node out of the grid to do whatever, and you folks are not even aware of it. And as it says avoid user or service disruption, in rolling maintenance that's really one of the strengths of it. And a little secret, every cloud you go into, whether it's Amazon, Azure, SAS Cloud, whomever. They’re all in the same type of architecture. Distributive processing, improved performance, meet changing demands, grow incrementally. And that of course that means we can add products and we are considering adding other nodes going forward for load. And we've done surveys over the years and it's pretty consistent that SAS is probably used, 8 to 9 times more than the rest of the analytical software combined. We book about 40, 44, 45 million processing minutes a year, and probably about between a 125 and a 134 thousand jobs each year, so it's very busy. Now that's not saying that things like R and Python are not equally important, MATLAB, etc., but at this time I even see folks coming in from academia who’ve used open source, who embrace SAS, and we encourage folks to use whatever tools viable, or whatever tool is useful. Leverage commodity hardware, reduce cost. That's true, hardware is actually one of the least expensive things in the modern analytical and production function.

Okay, now this is a spoiler alert. The next slide you see we had a long discussion with SAS about this. And these are some slides that SAS have provided to us, so if you see that they are somewhat of a more marketing nature then say I would provide or you would provide, you're probably correct. So one thing our current version of SAS does, and I've presented this many times before, and it does get a great amount of usage though I think it's going to get more usage as time goes on. We can call R packages from PROC IML. And you do not need to do anything special to call an R package, provided an R package is installed. And you can make a request to us, SAS admins, and we pass that along to our Linux Admins and they will download whatever package you can desire. This happens frequently. We can call library package from inside a submit block to load the package. You can then call the functions in the package. The example in this section calls an R package and imports the results into a SAS dataset. And we define the data and transfer the data to R, I'm also going to show this live a little bit later, call R functions to analyze the data and transfer the data results of the analysis into SAS, IML vectors. And there is, for your handout, there is a little link there for support that will show you how to do it, and offer a lot of valuable information.

Now, call R packages from PROC IML, here’s an example, this is very simple. Just making a simple define the data in the SAS/IML vector then transfer data R by using the ExportMatrixTo R subroutine. In R, the data is stored rather in a vector named rq. So very simply. Here it is, PROC IML, q equals yadda, yadda, yadda, yadda. RVar equals rq, call Export q, RVar. Okay? Now that’s from SAS. Then we load the KernSmooth package, I think that may even be an older, more obsolete package, at this point. Because the functions in the KernSmooth package do not handle missing variables, and non-missing values in q must be copied to matrix p. There are no missing values in this example. The Sheather-Jones plug-in bandwidth is computed by calling the dpik function in the KernSmooth package. This bandwidth is used in the bkde function in the same package to compute a kernel density estimate. So what they're doing, their submitting this, submit RVar R and you're essentially calling up, as many of you know, in R and we're processing it forward yadda, yadda, yadda. And then we send put EndSubmit because all these occur as submit blocks within PROC IML. What is actually occurring is that while SAS is controlling a lot of the memory and the IO you are passing control over to R and back. And then we copy the results into a SAS dataset or a SAS/IML matrix or whatever, and you can even do graphics. However I will warn you that graphics have to be saved, because of active X in the Linux environment, the graphics do not display themselves, but you can save them as a jpeg, or whatever you like. Copy the results in a SAS dataset or a SAS/IML Matrix and perform additional computations. For example, the following statements use a trapezoidal rule to numerically estimate the density that is contained in the tail of the density estimate of the data. And I'm not going to read it this off to you, you can see that, and it will be in your handout and again we have, where this comes from. And again will show you a live portion of this.

And next. We can use SAS to integrate R. Now what you are seeing here is actually what we currently have at Enterprise Miner. Now we have Enterprise Miner version 14.3 and I'll tell you that is an extremely popular software out in other parts of the government and in private industry and a lot of open source people really enjoy it and one of the reasons is because you can include open source modules in your analysis. And as it displays here, for example if you wanted to do a comparison logical regression R, logical regression SAS. Decision tree R, decision tree SAS. And what is really cool is, and this is used quite frequently and I encourage this, and I used this quite often, there's a model comparison and the parameters of the model comparison are derived by you. For example, if you were saying, and they're using home equity, which type of home equity program will glean us the most profit. And then define your model comparison which model shows, leads us to best probability and it will choose it for you. So it could choose any of these, and we can display that later. And we can model compare it, why? Model comparison, leverage R for new algorithms, generates score code, deploy R models. We have SAS models, we have R models. So you can use them in tandem or you can, for example. If there is a R model you prefer, that's not a problem you can put it out there, it'll pass everything back to SAS. If there is a SAS model you prefer, to pass in R, whatever combination you like it's quite flexible and quite powerful. And to proceed.

Now SAS from R. All of you people I'm sure recognize R Studio, all of you R users. And there are ways to call SAS for procedures, etc. into R, via here. And it runs this through, and I'm not going to read this all off, you'll have the handout. This is SAS plus R and it will give you your results, and show you nice graphics.

Now, let's look at a quick live demo of what we have just discussed.

And that I will take you to Enterprise Guide. Now, for example, here is a very simple way to look at it. We say PROC IML print this, submit R, we're just submitting some very simple R code. And our results, are here. Showing that they both work concurrently, they both display the same results. We can merge SAS into R, which you saw earlier, essentially run ExportDataSettoR. I created a silly thing called pets, and just send it to a data frame. Okay? So, you would say why would we do this? The weak point, to me and technology of open source, in academia it's very good because you're talking about very small datasets. In the VA small datasets really don't exist, unless you prepare them. I've seen people try to run 500 million rows through open source and really they can shut down a server. We even have to be careful within the SAS world on the SAS grid, that when people are doing very large jobs, we help them do a little bit of data preparation and it runs much more smoothly, but the tricky portion of it is, is open source is not really written by folks who, they’re written by very intelligent say statisticians, or subject-matter-experts. They don't put a lot of effort, usually though I believe it's getting better, into the actual optimization of the process.

And let's go here, and we have that of course. Now we do SAS to R data export SAS out to the data frame as we've shown before. And our results are, here, R, SAS. Now, here is a simple R demo. And this is one thing, I think a lot of people aren't aware of but are becoming more aware of, because SAS is a grid what is the power of any grid? The power to run in parallel, to run concurrently. Meaning if I have prepared my data and I wish to run various analysis against it, and I'll demonstrate the results in Enterprise Miner, if I adapt this towards a grid parallel processing and it's just as easy to do as by saying analyze program, analyze for grid computing, and it will essentially take this. And it's thinking a little bit. And I would say grid, begin analysis, and this may take a moment or two. And that's all it needs to do. And then I can add that to my project, alright? And it's come back up here. So what it has done, it's taking the R modules and actually split them up so they'll run in a parallel mode. Now I did this previously, take my other example, I just want to show you how easily it's done. And we would run this, and now these are very, very small light examples, so you're not seeing the true power of the grid. The more complex, the problem is or the programs are, the greater the ROI, Return of Investment, is for a grid. So if you're doing very, very small things like this, I would not necessarily put this in a parallel mode. If I'm doing something that takes an hour a piece, if I do it in a linear fashion I have a 5 hour program. If I ran it through the grid analyzer, I have a one hour program because it takes as long as the longest section of the concurrent processing, and it will give us the same results. And each of these was run on a different job slot, on a different node, okay? So that essentially is how we do it, and here, I'd like to discuss before what happens a lot of people have come back and said I can't get my graphs out, I go through the grid and an individual, a good friend of mine a very knowledgeable individual John Cashy, this is the method he uses which I have cannibalized. And he sends everything out to a jpeg. And what happens is you run a simple thing like a SAS Class, and you table jpeg it out, plot it, and you can get any graph there and you can send it back as you know either to your windows environment or to your Linux environment, copy back over with Win SEP, and the graphics are very, very good, they’re publication ready depending on what you choose. So, what have we displayed here? We've displayed the usage of R within our current version of SAS and we're going to actually expand upon this later. A lot of this is being done currently.

Now let's talk about Enterprise Miner a little bit. This is Enterprise Miner, this is a different tool. What this tool does, is you essentially set things up and everything is done, its parameter driven, pull a node out, drop it and then you essentially define the parameters of how you want things like I want a trading 55, validation 45% here. I can do replacement, things of that nature, imputation, I have many, many things appear. And this is version 14.3. In the next coming months we’re going to upgrade to SAS maintenance release 6, we are at maintenance release 5, and I’ve seen 15.2 and it is very well enhanced, very nicely done. But all you need to do is just click on a tab and it will tell you what these are. There's a very nice help feature up here and there things like my performance, in class file, data partitioning, explorer, GLMs, imputations so you can work in either mode. It has utilities where we can register a model, a reporter, SAS code and we'll talk about that in a moment, open source integration we'll talk about that in a moment, but I'm not going to go through each one because this would be a seminar in itself, and we have time series. Essentially it is a very, very powerful tool you do not have to be a programmer to use this you just have to understand what you want statistically and you have to understand your data. As we all know modeling and analysis is 90% data. So this is just a very simple example that actually comes with Enterprise Miner, but if I wanted to use R I'd go open source integration. And if I wanted to use Python it would be SAS code, and I'm just going to gloss over this briefly. And I could connect these to my control point and I could do a model comparison and in this case for example let's look at the results of a model comparison. In their example here, what they wanted was greater profitability of all of their statistical procedures, and we found that the interactive tree has our best rate of return, average profit. Now I could easily have added R modules here, which would come up, Python modules here which would come up. And by the same definitions it would show me which ones are the best, and I can and will adjust them accordingly. So here I have things like autoneural, regression, neural network, interactions, decision tree, the winner, decision tree. And I can put anything I want from R here, and anything I want from Python here, we just have to have it downloaded. And we can score it and we have SAS code and you could put other SAS code in if for example you wanted to sort it differently or bring it up in a certain way. So, our two main products are Enterprise Guide, and Enterprise Guide as I'm sure as many of you know, has a wonderful wizard in anything here. So again you don't have to be a programmer as much most people, I find, who are SAS programmers just like to go in the program window and program away. But between Enterprise Guide and Enterprise Miner you have a lot of power, a lot of analytical power at your fingertips, and one of the advantages of Enterprise software are the UIs, and these are big advantages in SAS as two excellent ones.

So let's proceed back to our demo, I'm sorry back to our presentation. And possible future. How can you open your analytics programs to all types of programming languages and all levels of users? How can you ensure consistency across your models? And SAS’s solution is SAS Viya. Now this is a new product that came out, I believe year in a year and half ago, and I know in private industry that that's being used quite a bit and it's being used quite a bit in the government too. We haven't got around to it yet but I'm really hoping for all our sakes that we do. And what is SAS Viya? Architecture that is enabled by wide ranges of capabilities and again as you can see here this is something that SAS put together, so, well take it for what it's worth. But what they're putting down here, none of this is untrue it's been proven out in the industry. End-to-end data cleansing data mining and machine learning process with a comprehensive, visual and programming interface. And that is very true, you can also use Enterprise Guide with SAS Viya and Enterprise Miner. There's actually, I should have shown it, but there's a node in there for SAS Viya code. Extension of the success of the SAS GRID @ VINCI, and SAS GRID has been enormously successful and this would make it more so. And today the VINCI Grid utilizes SAS via user interfaces such as Base SAS, hopefully very little, because Base SAS is essentially vestigial. When SAS 9.4 came out they stopped enhancing it, they published that, they're always going to give it to you but it's never going to improve they want you to things like SAS Enterprise Guide, and other much greater functional UI’s. So we have Enterprise Guide and we have Batch Submit which you know as gsub, and 9.4, and Viya. They can coexist very well and they enhance each other very well. It's an enhance current strategy Viya extends 9.4. That's very true because it allows us to more easily use R and Python and Java and many other types of open source. Each is designed to solve different use cases. They can coexist and R Grid is a perfect example of where that can happen in an ideal venue. Data, models and code can be accessed directly. And they're using sample M5 which you are currently, and 9.4M5 which is our version of SAS, it’s just another library engine.

Now here's some notes that I found that I actually want to include. SAS grid will continue to be needed to provide workload management of existing new customer SAS programs. You don't have to change a thing in Viya from your SAS programs if you choose only to use SAS. It continues to be the cornerstone for modernizing existing SAS environments. SAS Viya provides, among other things, the CAS server and that's Cloud Analytical Server. Now a cloud, in our case, if we bring this in we’re going to hang this off of our grids and our grids, in fact VINCI is a cloud if you really want to get technical about it, we will not be going out, which would cause essentially disrupt what VINCI is for, we would not be leaving the VINCI environment to go out to a cloud with any set of data. All data will stay within VINCI, Okay? And the CAS server, an in-memory, distributive run-time engine, and we’ll talk about later, to enable visualization, deep learning, machine learning, etc., against very big data. A combination of the Grid and SAS Viya can work with the type of code that VINCI has at the VA. Using SAS Viya to extend a SAS Grid environment becomes seamless 9.4M5.This will have CAS client connectivity within our grid. Do everything from our clients, 9, 4, and 5, run all existing grid jobs without modifications. Run CAS engine, CAS statement, CAS enabled procs by adding them to existing SAS Grid jobs or creating new jobs. Which means we can combine them seamlessly, and that also means because that can run R and Python, etc., you're combining R, Python, within this SAS Grid seamlessly. You're not just running it from a single server. SAS Viya and SAS 9 is an and strategy. Each is designed to solve different use cases. With the latest releases, you can access SAS Viya processing from any SAS 9 coding client, like SAS Enterprise Guide, etc. No longer is SAS CONNECT bridges needed, meaning that SAS CONNECT is something you don't see in the background, it connects node A to node B, etc., and combines and chops slots. Many of the SAS 9.4 products have direct coding connections within workflows as well, like SAS Enterprise Miner, SAS DI Studio, etc. As we had seen earlier.

Now let's proceed. Now this is definitely marketing. Open Multiple Interfaces Single Code Base. So if you look here you can, from a single environment access all of these, you have visual interface, you'll have Python, SAS, Java, R, and more. API Interfaces, Rest API and you can work it all under one venue. You don't have to pop out the R Studio, you don't have to go to Linux, and run Python, you'll have to come to the grid to run SAS, etc., etc. We’re running everything under one venue under the SAS grid which is our most powerful venue.

And the analytical life cycle. Discovering and Development of Analytics, which you all know pretty well I'm going to assume. Deployment and Execution of Analytics, we've prepared data, we explore our data, we model it, then when we deploy it, inventory, execute, monitor, etc. Wishlist, what everyone wishes for, best possible analytics, flexibility of tools, productivity, greater insights equal models, trusted models. How does SAS extend, not only SAS code or what we do there but the other inclusions of open source and other technologies? A variety of options to develop models, allows data scientists to code in a language of their choice, ability to scale to any data volume, handle complex graphics. Now, the ability to scale to any data volume is very important to us in the VA. As we know our data is, we’re petabytes, I mean there's terabytes involved, but we’re in a petabyte era now and I'm sure within the next, as things will advance geometrically, especially whether EHRM it's going to start to become exabytes.

SAS as an enhancement, what we’ve said earlier Python, Spark, Lua, Scala, R and SAS. SAS can consume all of these under the one venue, you can work in whatever language you prefer. As they say SAS can augment open source, increase productivity, leverage your assets, people and platforms, bring the power of SAS to open source, create deployable analytics, and for example in SAS Enterprise Miner you have model packages. Once you're done you can put the model out and it can be used, we could put it as some people do into a stored process and anyone can use it at that point. Goal is to embrace and extend, and that essentially goes back and forth. Open source to SAS, SAS to open source.

And again it's exactly the same. Analytical life cycle SAS and open source, Discovery and Development, what we saw earlier. And SAS combined with open source, SAS embraces open source for data prep., though I honestly think data prep. is a little better in SAS but that's because of the machinery behind it. Open source and SAS work well for discovery and development, I’ll agree there. SAS can extend open source, and vice a versa. Inventory, register and managed models, deploy execute models in Hadoop and in databases, enhance models and provide monitoring and reporting, embrace and extend.

Now you ask yourself how can this be done, How can you include everything? What SAS has developed is something called the SASHDAT file formats and what that can use, and I have a white paper that explains a lot of the architecture of SAS Cloud and essentially Viya, but essentially what it is, it takes any data you have and puts it in a very readable format that can be shared among any of the open source, or any program that we have installed there. And it can handle things like CSV, SAS Space, any database access, Oracle Teradata, you name it. SPSS, Excel, JMP, a format that you specify and it works very nicely with Hadoop and Hive. So the need to unify analytics assets, Analytics Governance and at this point we really don't have too much of analytics governance. We have the SAS Platform, we have some people doing open source, but it's probably not as structured as would be needful. Precision results you can trust. User appropriate interfaces and interfaces and the SAS world are very good, but R Studio is not bad, but Python is pretty much just straight away coding and you have to be quite a knowledgeable programmer to use Python effectively. Analytics for the masses, meaning it encompasses far more folks, that doesn't mean unwashed masses of course. Streamlined Deployment, Effective Processing. So when they say streamlined deployment and effective processing happened under one powerful venue managed by SAS Admins, Analytics Admins in the back end and analytics OS is ideal for high performance in the analytical world. Windows, not so much. Windows is token based, wonderful user interfaces desktop etc. but in high speed world you're not going to see a lot of windows products that are going to compete. Scalability for large complex or time sensitive problems again that comes back to parallel processing.

And moving forward we have open consistent results across different interfaces. Here's Python as you folks who use Jupyter Notebook, can see, and there's SAS Studio and the same would happen, actually if you use SAS Studio, you can use Enterprise Guide, you can use Enterprise Miner and make calls to SAS Viya, which means you can include the open source. So, this would you could get the same results, I'm sorry under SAS Studio using Python or using R. This is Model Studio which is a product of SAS that we do not have showing, just verifying the results.

And open source code base example. I was showing SAS and you have everything, everybody recognizes proc print, Python folks will recognize Python code, R etc. Very similar. How it goes into the controller, and our controller is would be the CAS server and it'll put it out to the worker nodes which would be the grid compute nodes. And they communicate back and forth as we do now between SAS metadata and the compute nodes.

SAS and open source, again. 9.4 SAS Viya and then they had to put this one in embrace open source by including it and leveraging it where we can. Extend open source by improving it's interoperability and utility for the Enterprise.

And I've included many, many on the handouts there's a lot of documentation. Like empowering the SAS IML user with the functionality of R. I welcome you all to take a look at this, we have videos, videos we have blogs, many, many papers, articles, and if you go out and Google you'll see much more than this.

More information extending R, using R in SAS Enterprise Miner, more blogs, Spectral Clustering in SAS Enterprise Miner using open source integration node, all very interesting. How to execute a Python script in SAS Enterprise Miner. I've always said SAS Enterprise Miner even though every year it's more and more consumed is still underutilized. I don't think people appreciate the power of it until they start to use it. Once they use it people are in it whole hog. Most anyway. And then we have open source integration using a Base SAS JavaObject, if that's your preference. And the open source integration node installation cheatsheet and then we have usage notes from SAS also.

And for more information materials on GitHub, which we all know and love, SAS integration sample code, Integration with R and Python. GitHub integration with Jupyter Notebook in Python, using SAS on GitHub. SAS machine learning methods. SAS Enterprise Miner process flow diagrams. And read this, this is all very interesting if you have the time and you're interested you're going to glean a lot of information by reading this.

And resources we offer internally as you all know the VINCI SAS Admins offer online training for you or your group. We have an excellent trainer, Tony Sullet [phonetic] Who that is his primary function here. He's doing a very nice job. I hear accolades in that regard. And you can contact us at the usual areas VINCISASAdmins@va.gov for SAS, or VINCI@va.gov for access, network, data or other questions. For example if you can't get into VINCI to use SAS sometimes we can log you out of the server but normally that's a VINCI question, it's more networking it's not us. And there's our VINCI central site that has a lot of wonderful documentation, the vast majority of it written by Kevin Martin who many of you know. Who is one of our valuable SAS admins, very much program and user oriented.

And then we say to you good gridding, and I want to thank you for attending. As always, a pleasure to present to you. I enjoy it. I enjoy speaking to you as I hope you enjoy listening. And please contact us with any questions or comments and I can open this up Robert to questions, we have about seven minutes.

Robert: Right, we have one question pending. And people can use the questions pane if you have them but yeah like not an awful lot of time like Mark said. First question, do we have access to SAS data management software for ETL?

Mark Ezzo: Yes, we do. And that's primarily on SAS Enterprise Guide there's a lot of wizards for it or SAS code if you know how to do that. If you have any questions contact us and we’ll be glad to set up some training.

Robert: At the VINCI SAS admins at VA dot gov address?

Mark Ezzo: That’s correct, as on the screen right now.

Robert: Well that was the only question that we had pending. I'm surprised that there aren't more, people are probably typing right now\_\_

Mark Ezzo: Well I\_\_

Robert: Come flooding in.

Mark Ezzo: Many of these, I try to cover as much as I can, so it may be that they were, were I they I may ask a simple question, I would probably go look at a lot of the documentation first and come back with questions. Normally when I present a seminar I get a lot of questions after the seminar is over when people sit and think about it or try out a few things.

Robert: Right well I guess people are digesting all the information. There are no questions at this time.

Mark Ezzo: Okay, well if that's the case I can return about five minutes back to your life.

Robert: Sure, it's up to you if you have closing comments or anything like that?  
  
Mark Ezzo: Oh just very simple comments. It's very enjoyable from my aspect to help you all in what I consider very important research for Veterans and for humanity itself. I think that this is really one of the reasons that I get up in the morning and go to work with all of you folks. I enjoy helping you with your projects, help you do things most efficiently working with you with modeling as we all do in the SAS and VINCI environment. And we've been growing every year and I hope we continue to do so and with the new Cerner project coming in I think there's even going to be more opportunity to do a lot more important analytical work and modeling and we all look very much forward to it. And if you have any questions at any time about SAS or analytics or modeling in general please feel free to contact us. We enjoy hearing from everyone. And with that, thank you all very kindly.

Robert: Thank you Mark, people when I close the Webinar a short survey will pop up. I just want you to know we do look at those answers so please take the time to fill them out. And with that I'll just say thanks everybody and have a good day. Bye Mark.

Mark Ezzo: Take care, buh bye.

[ END OF AUDIO ]