>>Mollie Putzig: – about a minute here to give people some time to get logged in and set up.

Okay. We're gonna go ahead and get started here. So, hello, everyone and welcome to today's M&S series webinar: "ProteusLib, A New Process Modeling Library for Water Treatment." I'm Mollie Putzig from the National Renewable Energy Lab. Before we get started I'd like to go over a few housekeeping items so you know how you can participate in today's event.

During the webinar attendees will be in listen-only mode. You can select the audio connection option to listen through your computer audio or dial in through your phone. For the best connection we recommend calling in through a phone line. You may submit questions for our speakers today using the chat panel. The chat section appears as a comment bubble in your control panel. You may send in questions at any time during the presentations and then we'll have an open Q&A session at the end of the presentations. During the open Q&A you use the chat function or you can raise your hand using the "raise hand" feature as you see on this screen and I will unmute your line so you can ask your question. We are recording today's webinar. Without further ado I will pass it over to Steve Hammond.

>>Steve Hammond: Thank you, Molly. I'm Steve Hammond, also from the National Renewable Energy Lab. I lead the modeling and simulation topic area within NAWI Water Hub. Welcome to our webinar series where we are highlighting expertise, capabilities, and the latest computationally driven research related to water treatment. If by chance you are interested in giving a webinar yourself please let us know. We're always looking for speakers.

Looking ahead on Monday April 5, Professor Ilenia Battiato of Stanford will be our speaker, and following that on Monday May 3, Aaron Wilson of Idaho National Lab will be our speaker.

Today it is my pleasure to introduce David Miller, Tim Bartholomew, and Markus Drouven of the National Energy Technology Laboratory and Deb Agarwal of Lawrence Berkley National Laboratory as our speakers. Together they are leading an important part of the NAWI Water Hub, the development of an open source library for water treatment-specific models to predict the performance of water treatment processes, treatment trains, and networks. The topic of today's webinar is ProteusLib, a new process modeling library for water treatment. David, it's all yours.

>>David Miller: All right. Thank you, Steve. And thank you, everyone for joining us today. It's a pleasure to be here to have a chance to talk to you about some of the work that we're doing as part of the NAWI Alliance.

So, let me just kind of quickly run through the outline of the presentation. So, we're going to start off talking about what ProteusLib is and what it's intended to do, the types of questions that it's designed to answer, and then briefly discuss the foundation for

ProteusLib, which is a computational framework called IDAES. I'll then go through some of the inner workings of ProteusLib and talk about some of the analysis enabled by ProteusLib and get an example of some analysis that we've recently done. And then, we'll continue the discussion talking about some of the opportunities for leveraging ProteusLib, and then finally wrap up with talking about its availability and user and developer support and ways that the water community and especially the NAWI community can become involved in the project.

So, let's start off with the question of what is ProteusLib and what is it intended to do? And to do that let's think about water systems. And really, you need to think about water systems as being complex interacting operations. So, if we take an example of seawater that we wanted to desalinate in order to have a product water that we can use for a number of different applications, we will want to think about how do we design the system to do that so that we can have our seawater coming in and our product water coming out? And we know we want to have desalination, and desalination itself is a fairly complex process with a number of different operations operating at high pressure making use of membranes. But before you can even send that seawater to the desalination train it needs to go through a pretreatment process, and that pretreatment process is going to depend upon the specific qualities of the seawater as well as the specific requirements of the desalination train itself.

And then we still don't quite have product water yet. It needs to go through a posttreatment process. And anything – time that we change something in this process it's going to affect the rest of the process. So, if our seawater quality changes that might require a change in our pretreatment process. If we want to substitute a piece of equipment or a new technology somewhere that will affect some of the existing technology that's a part of that.

So, really, as we think about just a seawater desalination system there are a number of different types of questions that we might want to ask and different types of decisions that we might want to make. And so, if we're operating a desalination plant we might have a lot of decisions that we want to make around process operations and what can we do to handle variations? What can we do to [inaudible due to audio distortion] and potentially substitute in new technology? If we're starting fresh we might be wanting to ask design questions. How are we going to design a system that takes advantage of the latest advancements in technology in order to provide a specific amount of water over a period of time in a specific location taking into account the regional variability that we're going to _____ the plant.

We might be interested in making decisions about technology selection. What types of reverse osmosis membranes do we want to pick? Do we want to consider changing out our existing technology for a new type of technology? And we might also be interested in making decisions around R&D priorities, for example understanding existing technologies and existing approaches, looking at the pinch points and where there are the greatest opportunities to improve the overall process.

So, as we think about these different types of decisions we can also think about different approaches for making those decisions. And one way that we can approach decision-making is to make use of approximations. These might be considered back-of-the-envelope calculations where I know, for example, enough about my process and how it interacts that I can simply say, "Well, perhaps I need to add more chlorine at this early pretreatment step and that can resolve some of my problems."

We can also think about heuristics as another approach for making decisions, particularly for something like seawater desalination where there's a lot of experience with designing and operating these system, these rules of thumb that come out of long-term experience with technology and the way it works and the types of things that don't work. Heuristics can be a reasonable way of approaching decision-making and narrowing down the realm of possibilities and at least getting you in the right ballpark for the types of decisions that you need to make.

Spreadsheets are ubiquitous and widely used for decision-making. And if we're generally making routine decisions where we have a limited number of options and a lot of experience it can be a useful way of making decisions. However, as we start having more complex systems we might want to think about simulation. While there might be an upfront cost of building the simulator or making use of the simulator to begin with, these types of approaches to decision-making or decision support can help us understand more complex interactions or for systems.

And finally, we could consider optimization, which is really just simulation run multiple times or being able to change more variables at once to be able to achieve a desired outcome, and this could be especially useful when we know we want to, say, minimize the levelized cost of the water but there are so many variables that we need to understand how many of them work together in order to affect the cost for a complex system.

So, NAWI itself focuses on innovation for water systems, and the graphic here on the left shows the A-prime approach that NAWI is taking to develop innovative approaches for water treatment, as well as in the background a number of the different applications that NAWI is focused on. And if you look at these you'll see that it talks about distributed desalination and reuse, mobile modular treatment, and clean brines for reuse, grid-responsive distributed brine desalination, process innovation, and intensification.

So you'll see that NAWI focuses a lot on new operating paradigms. Distributed. Not the large plants that we're used to but smaller plants that might be geographically dispersed. Grid-responsive. That essentially says a dynamic process that's going to be varying the way that it's operating based upon the requirements of the grid. These are not the typical approaches that we think about when we think about water. Also as part of A-prime, new treatment technologies. Innovation and process intensification. New materials and approaches for treating water. So, technologies perhaps that we're not used to understanding how they're incorporated into a water train. And then, finally, multiple sources of water. So, going beyond seawater desalination, looking at produced water, for example, that may have variable quality over time, and so we need to think about how do you design more robustly and/or design a water treatment train that enables much more rapid reconfiguration in order to respond to changes, perhaps as it's mobilely being relocated for use at different locations.

And so, all of this innovation that's under – happening at NAWI requires new decision-making tools. And specifically, ProteusLib helps to support this type of water innovation that's an important part of the hub and its development. Specifically, it enables the optimization of complex interacting systems. So, when we're looking at innovative systems where we don't have a lot of experience, our back-of-the-envelopes, our heuristics, and even our spreadsheets aren't going to help us as much as we need them to in order to help us broadly consider new design and operating possibilities.

And so, for example, if we consider a simple desalination system where you might have 50 to 100 variables that are both design variables and operating variables and we want to understand how changing those are going to affect the levelized cost of water, even if we have a nice simulator we can at best vary kind of two decision variables in a simulation in order to understand how those affect the levelized cost of water, as represented here. But there's another, say, 48 variables that we didn't capture in this simple sensitivity approach. By looking at optimization we can optimize all the variables as a function of, say, feed concentration, which is an input to the system, and water recovery, which is our output, in order to determine how the combination of these two, with all the other variables being optimized, can affect the levelized cost of water.

So, our open source library, ProteusLib, is designed to accelerate water innovation by letting us look broadly at design spaces. So, the thing that enables us to build ProteusLib is a tool called IDAES. And it's the foundation on which we're building so that we can focus on the water-specific library of capabilities. So, the IDAES Integrated Platform has been previously developed and it exists – it is built on a tool called PYOMO, which is an algebraic modeling language that lets us represent algebraic and differential equations that can represent real systems. PYOMO itself is built on Python, a widely used, easy-to-use programming language, and it links with a wide variety of commercial and open source solver packages that solves the math problems for us.

So, within IDAES itself there is a modeling framework that lets us build models for different types of technologies and different processes. And this is the key aspect of IDAES that we're leveraging in order to build ProteusLib. And so, this represents in a hierarchical matter steady state and dynamic processes.

With these models, then, we can look at optimizing processes, and we have the tools and capabilities within the IDAES Integrated Platform to do that design and optimization. We can look at conceptual design, which helps us answer questions and configurations such as how many parallel membrane models does it make sense to use? How can – how should I best balance the capital cost and the operating cost? It has capabilities for looking at process dynamics. How does the process change over time? How can I control the process, particularly if I want to be able to ramp it up and down as, for example, grid pricing changes and I want my distributed water treatment train to be responsive to the grid? It includes capabilities for uncertainty quantification. So, thinking about things such as robust design and making sure that the process is going to operate under a wide variety of different conditions. And it enables data reconciliation to validate the models and to be able to take the models and use them to understand the way a specific plant is operating.

IDAES is open source. It's widely available. And it's been developed by a team of researchers from multiple national laboratories and universities. This is a picture of the team at our last face-to-face meeting, about a year ago. We have a wide variety of stakeholders that are part of IDAES that includes industry, it includes researchers from academia, as well as a variety of national laboratories. So, it has a very large user community and it is widely well-supported. And finally, it received recognition with an R&D 100 award last year. So, leveraging this integrated platform lets ProteusLib achieve its goals to help support water innovation.

And just to provide some examples from outside of the water space of how IDAES is being used, we'll take a look at some recent applications and impact. So, for example, there's – in a design space exploration for an energy-intensive process, it was able to demonstrate that it could reduce energy demand by more than 42 percent by the automated exploration of over 42 million alternatives. An example of a specific carbon capture application it reduced the operating cost by 18 percent. It's been able to identify robust designs in the context of a carbon capture technology to ensure that that design was robust against uncertainties and could still achieve a minimum capture rate. In process improvements within power plants it was able to de-bottleneck a plant and – so that it could reduce its minimum operating load by 44 percent and increase the overall efficiency.

IDAES is being used in multiple other projects, including DISPATCHES, which is a product that's part of the Grid Modernization Lab Consortium, that is looking at the design of tightly coupled hybrid energy systems. It's being used – it's a part of an ARPA-E project on flexible carbon capture. It's being used through CRADAs with multiple industry partners, including design and optimization of direct air capture systems. And it's the foundation for a number of projects with small businesses with funding through Small Business Innovative Research program. And now, most recently it's the foundation for NAWI's ProteusLib water treatment library.

And with that I'll hand things over to my colleague, Dr. Tim Bartholomew, who will provide a deeper dive on the inner workings of ProteusLib.

>>Tim Bartholomew: Thanks, David. Took me a little bit to get to the unmute button. I hope everyone can see my slides. I'm going to be talking about the inner workings of ProteusLib. And so, with that, it starts off with what David was mentioning, the IDAES platform, which is represented by this yellow box. And I just pointed out the key features. It's open source, equation-oriented, has an extensible library, it has a fully featured programming environment in Python.

And so, on IDAES we use the unit operations and you assemble them into flowsheets, and then you can have some capabilities like simulations, optimization, dynamics and control. And the key part to be able to use these is you need to build these unit operations. And these are made up of unit models, reaction packages, and property packages. I'll give you a little bit more detail behind these. Unit models would have balance equations like mass conservation and energy conservation, and also performance equations for that process unit. They also have to be paired with property packages to calculate the properties that are used in our model. These have ideal property models, equilibrium equations, and mixing rules.

And so, where ProteusLib fits in this is that it is the collection of unit models, reaction packages, and property packages that are water treatment-specific so that we can do all the other capabilities, the simulation and optimization, with water treatment processes. So, to describe this a little more I'm going to walk through the one unit operations example of reverse osmosis.

So, reverse osmosis, it is composed of both a unit model and a property package. And as I mentioned before, unit models are things like mass and energy balances and performance equations like the flux calculation or pressure drop. And for ProteusLib we are going to build multiple levels of detail – so, a hierarchical model library. And one of the first basic levels can be a zero-dimensional RO model with really simplified relationships, like assuming a pressure drop or things of that sort. And then we'll also have advanced models where they're one-dimensional and reflect the profile across the membrane stage and include detailed relationships like estimating the pressure drop with friction factors that are based on the Reynolds number or for a specific mesh channel or concentration polarization on local mass transfer coefficients and things of that sort.

This also has to be paired with a property package that determines property relationships and equilibrium equations. At the basic level we could consider a single component, like sodium chloride, or – and that would be mostly just simple regressions. For example, I'm showing here osmotic pressure as a function of mass fraction and temperature. And there could be other properties like this, like viscosity or density, for this single component that could be represented by polynomial expressions.

We can also have more advanced models that are multi-component and theory-based, like the electrolyte non-random two liquid model system, or other types of property packages that represent electrolyte systems, which are fairly complicated because they have lots of true species. There's lots of different ions that can be in the water systems. And then, also a parent species, like the solid – the salt ion pairs and things like that. And so, ProteusLib will support these types of property calculations as well. Like, the main outcome of electrolyte non-random two liquid model, we use the activity coefficient. And you can use that for precipitation or also calculating osmotic pressure and things of that sort.

In a similar line, I'm showing another example for chemical treatment, which is heavily used in pretreatment and posttreatment in water treatment systems. And a unit model would be a chemical reactor, which can have a variety of level of details. Again, we can be stoichiometric-based, equilibrium-based, kinetic-based. And all of these unit models are already developed on the IDAES platform. And it's the core model libraries that has chemical reactors.

So, the ProteusLib contribution is the detailed property packages and reaction packages for these electrolyte chemical systems, which as we know can be pretty complicated. So, the property packages include property relationships, physical property parameters, and phase equilibrium, where the reaction packages provide the system of equations for the reaction and the data backing up those reactions.

So, I hope that gives a good idea of what the type of models and what their underlying structure is. But I also want to place the context of ProteusLib within NAWI on the spectrum of models ranging from simple to complex. Within ProteusLib we will have a range of hierarchy, but just to do a little even broader reach.

So, the simplest model, a zero-order model, this is something that doesn't predict performance. It's not physics-based and it assumes the performance, like the recovery or water flux.

Zero-dimensional models are physics-based and they can estimate the water flux but they take significant simplifications, like assuming that the average water flux can be based on an estimated water flux at the inlet and at the outlet, and then that can be used to estimate the water recovery.

One-dimensional models are more detailed. They really lay out the whole profile of the water flux and all of the state variables across the membrane stage.

And then, we also can get to two- or three-dimensional models, which are very complicated and are representing fluid flow in a mesh-filled space or channel. And unfortunately, these very complex models are too computationally intensive to be able to use at the process scale in a flow sheet type of simulation model, so what ProteusLib particularly focuses on are these zero-dimensional and one-dimensional physics-based modeling approaches. These link the decision variables to the performance metrics, which we can then relate to optimizing the levelized cost of water or energy consumption and those types of outcome metrics.

These ranges that we don't address as much, for example the very simple order models, those are more aligned with Water TAP[3] – it's another team within NAWI – their techno-economic modeling platform approach. And so, they use very simple models but detailed cost models in order to estimate the techno-economic feasibility and screen technologies. We are currently working with that team to integrate those models into the ProteusLib framework and make them compatible to be one integrated tool.

On the complex side of things, this is another sister project to us, ProteusHF, which stands for high fidelity, which includes doing computational fluid dynamics for the fluid flow in mesh spacer channel – mesh-filled spacer channel where it can calculate mass transfer coefficients or friction factors. And we can reduce those relationships and make a reduced order model and pull them into the ProteusLib one-dimensional models, for example, to try and replicate those calculations.

For the big vision of ProteusLib, this is – the main thing is it's a water treatment library. So, I'm just going to present a few of the things that we have in the immediate plans and through the rest of the NAWI project. This includes – we're primarily focusing on RO desalination treatment trains within our first year. And so, we use cross-flow filtration, which has RO and nanofiltration; dead-end filtration, with ultrafiltration, microfiltration, and porous media like sand filtration; different chemical treatment processes; and auxiliary equipment. And then, these are paired with different property models that can be the simple sodium chloride progression-based models up to equation of state and then electrolyte-based models.

In future years and with the help of the NAWI community and other projects developing models we're hoping to add things to the model library like counter-flow membrane processes, evaporative processes, and electrochemical processes. But this sort of rounds out the models that we have proposed for the library. And what we're hoping to do is use these models for simulation and optimization on a flow sheet basis, but also to leverage the advanced capabilities that the IDAES platform provides. And so, this includes things like dynamic modeling, which David emphasized earlier – so, we can look at time-dependent process, like closed circuit reverse osmosis; or fouling and scaling; or variable feed conditions. Network and conceptual design, which identifies promising novel designs – and we'll talk about this later on when my colleague Marcus is talking about different network designs that we'll consider going into the future. And then, also surrogate modeling. And so, the idea is it already has built-in tools to take in data from either experiments or from simulations from ProteusHF, the high fidelity modeling, in order to turn it into reduced order models that are more applicable for a process flow sheet analysis.

And so, hopefully that gives a good idea of the vision and the breadth, the scope of ProteusLib. And just to emphasize how it works, I'm going to dive into an analysis for a specific process.

To start, though, the main core function of ProteusLib is to provide predictive analysis to help a decision-making for early stage applied research. And these types of things could be estimating performance metrics, like the levelized cost of water, energy consumption and other pipe-parity metrics; extract design and operation guidelines; identify bottlenecks; and estimate a range of outcomes. So, incorporating uncertainty into these processes.

And so, the results could be simulation-based, where we're varying two decision variables and seeing how it affects an outcome metric. But primarily we're focused on optimization, where those decisions variables are optimized for some objective like minimizing the levelized cost of water, and then assess the technology over its full application phase, with feed concentration and water recovery.

We also could do sensitivity analyses and see how the outcome metric is affected by changing a parameter, and then more of a stochastic approach, where you include uncertainty for a variety of parameters and get a distribution for what the expected outcome metric would be.

And so, I'm going to walk through these types of analyses for a specific process, and in particular this relatively newly proposed technology by Wang et al. at Columbia – or, not Columbia, at Yale – for low salt rejection reverse osmosis.

And so, this process has a multi-stage process, and the first stage is always reverse osmosis, and it produces the product water. And this is your typical RO, maybe even seawater desal type of reverse osmosis. And the concentrated brine is then fed into this low salt rejection reverse osmosis process, which can concentrate the brine further. And the permeate is recycled back to the RO system, and then the concentrated brine is disposed of.

And to describe the reasoning for this process I'm going to just touch base on the driving force for reverse osmosis, which is to drive the water flux, it is the water permeability coefficient multiplied by the driving force, which is the applied hydraulic pressure difference minus the osmotic pressure difference. And so, in reverse osmosis the feed side is – has the saline feed, and so it has some sort of non-zero osmotic pressure. But then, the permeate side is almost pure water. It has very little salt content and negligible osmotic pressure. So, what we do is we apply a high hydraulic pressure and have a – and there's some atmospheric pressure on the permeate side, and we can apply a large applied hydraulic pressure difference. And that can overcome the osmotic pressure so we can have water flux.

But as the feed concentration increases the osmotic pressure also increases, and it can go to a point where it can offset our applied hydraulic pressure difference. And we have a few options here. We could potentially increase the hydraulic pressure that we're operating at, but typically we're already operating near the maximum that the membrane modules themselves can handle. There is active research into high pressure reverse osmosis, but it's an ongoing area to try and raise the operating pressure that we can operate at.

Alternatively, which, well, few emerging membrane processes do, is they allow there to be a saline – there's some salinity content on the permeate side. And so, there can be an osmotic pressure on this side and it reduces the osmotic pressure difference so that we can still have water flux for our typical applied hydraulic pressures. The way low salt rejection reverse osmosis does this is it allows salt to pass through the membrane. That's within the main as well as salt reduction, so it's allowing a significant portion of the salt to pass through so that there's osmotic pressure on that side. And so, with that you can still drive water flux even at a higher salinity, and then eventually concentrate brine more towards disposal.

And this process, it doesn't just go up to very high salinities with a single stage, but you can get to mere saturation for a sodium chloride solution by doing multiple stages in series. So, here is showing an n-stage LSRRO process, and as you can see mostly you do an RO process up front and then it feeds into an LSRRO process where the permeate is recycled back to the previous stage, and it just steps down from that.

One of the things that's complicated about doing this type of multi-stage analysis is there's a lot of – in order to do this a modular flow sheet is really useful because the person who is assessing the technology can use a model library and just connect things to each other instead of creating each thing individually. And so, in a flow sheet this can be rapidly assembled.

Additionally, another thing that a user is challenged – that is challenging for a user is that there's a lot of degrees of freedom. So, for each stage there is at least these key decision variables: what the membrane area should be, what the operating pressure should be, for the low salt rejection reverse osmosis if we can tune the salt permeability and its passage what that should be. And so, there's a wide range of degrees of freedoms. And so, handling this is usually pretty difficult, especially when you don't have experience operating these processes. And this is just a theoretical – this is not a process that has been applied out in the real world yet.

And so, generally what people do for a simulation-based approach is they vary those decision variables. And so, for example, you have your model and you provide the case specifications, the process and operating conditions. You also provide things like the membrane properties, process, and financial parameters. Then you solve it and then you can get the outcome metrics like the energy consumption, the levelized cost of water, and the water flux. But the key thing here is we don't – there's a big uncertainty about how the process should be designed and operated. And they're very closely interlinked because there's all the recycling and like that. And so, they have a significant effect on what the outcome metrics should be.

And in traditional analyses oftentimes we don't set on a specific design and instead two decision variables are varied and what effect they have on the outcome metric is plotted. And this is – and you can do this for when you're varying one to two design and operating inputs, but for low salt rejection reverse osmosis processes we have three degrees of freedom for each stage, and so it's not very manageable to do it with this type of simulation. Also, they're very interconnected, and so it's difficult to do what in a typical analysis would do would be fixing the other decision variables while leaving them _____.

So, the better approach is to use an optimization approach where these decision variables are separated from the inputs. So, now they are part of the model, and an optimization solver would determine what these decisions variables should be given an objective like minimizing the levelized cost of water. And if we do this type of approach we can do what I shared before, where instead of varying decision variables now you vary case specifications, like feed concentration and water recovery, and you can plot an outcome metric. And this is really – expands the analysis greatly because now we can assess the technology over its full application space.

So, here's the results for low salt rejection reverse osmosis. I'm showing the same axes: feed concentration on the x-axis and water recovery on the y-axis. And over here on the left is the energy consumption and on the right is the levelized cost of water. The first thing you can notice is that the energy consumption and the levelized cost of water vary significantly across the domain space. And then, additionally I'm pointing out the cases where – which design is optimal for the given feed concentration and water recovery. So, at the low salinities one stage is optimal, and that's just reverse osmosis, the LSRRO process. But then we add those recycle stages and LSRRO modules and we can have higher concentrations and things of that sort.

Just to give a benchmark data point, our cost that we estimate for seawater desalination at 35 parts per thousand and 50 percent water recovery, we estimate that to be about $0.75 per meter cubed, and that doesn't include cost associated with pretreatment or posttreatment. It's just the desalination treatment step.

And so, this analysis is really useful for guiding what we can expect for the levelized cost of LSRRO and how it can expand the treatable salinity. And these plots are made up of hundreds and maybe even close to a thousand individual optimization runs. And the IDAES platform is very efficient at handling things with a high degree of freedom. These simulations individually take only about a second, so to generate plots that are like this are on the orders of minutes.

And so, if we move on, one of the reasons why IDAES and a ProteusLib framework is really good at doing these efficient optimizations is that we use an equation-oriented approach, which has significant benefits over traditional optimization methods. First – most people, I believe, are familiar with traditional optimization methods, so I'm going to describe that first, where we have a simulator where this is your process models, your energy balance and mass balance, your performance equations, and if you provided it, the decision variables that would tell you what the outcome is. And so, this usually gets paired with a separate optimization routine where you are minimizing the objective function for these decision variables u. And then, these u decision variables have bounds, and the optimization routine sends the decision variables and calls the simulator and you get this output. And it iteratively does this process. And for this black box type of optimization that is derivative-free operation, it takes about 1 to 1000 simulations for just a small number of degrees of freedom.

The simulator can be thought of as just a system of equations that is represented like this, and an equation-oriented modeling approach combines these two things to minimizing the objective function and it's still exposed to the full system of equations and all the intermediate variables and the decision variables. And this has a thousand-times speedup. This glass box optimization is a thousand times more efficient than this black box approach. And so, that allows us to be able to do these multiple thousands or optimization runs for a process in relatively quick time, even on a personal laptop.

And so, I'm going to provide a few more analyses besides just the maps that I provided before. Because we're optimizing everything we can plot the decision variables instead of just the outcome metrics in order to extract some guidelines for how to operate the process. So, here we have the optimal – cost-optimal membrane area for three different case. These are all higher salinity desalination cases than reverse osmosis can do by itself. And what you notice is – well, first of all, identifying what the membrane area – it's pretty complicated. It's heavily evolved in all the recycle loops, so the default passage and the water recovery – so, it's interconnected across all the stages. It also is a key component in balancing the operating and capital costs. So, it's not intuitive to identify these things but the optimization model can synthesize these relationships and determine what the value should be.

And so, for these three different cases we have a different number of optimal numbered stages, but these are some sample results. We also can show the salt passage and we see that the RO stage has very low salt passage, which is what we expect. But then, the LSRRO stages have significant passage, and we see that this is much higher for the higher salinity cases.

In addition to just plotting decision variables we can also vary parameters to help us guide technology development. For example, here I'm showing the change in two parameters, like membrane cost and water permeability, and its effect on the levelized cost of water. The circle here is our base case condition, but there's active research in improving membrane properties. So, if researchers are increasing the water permeability they can see how that would impact the cost. Usually, new novel materials cost more, and so we can see if they both increase if it actually reduces the levelized cost of water for the full process and can establish guidelines on the limit of membrane cost increase for some increase in water permeability.

And so, with that I'm wrapping up the analysis example and I'm going to hand off the presentation to my colleague, Marcus Drouven, who is going to talk about more than just our core analysis but also opportunities that we can use to leverage ProteusLib elsewhere. Marcus?

>>Marcus Drouven: Thank you, Tim. And Molly, I'm still waiting to be able to share. It has not come up yet. Hopefully, that will happen momentarily here.

>>Tim Bartholomew: Maybe, Marcus, I might be able to hand it off to you.

>>Marcus Drouven: There we go. I think I'm able to share my screen now. All right. So, yeah, Tim just talked to us about how we can use ProteusLib to do some really nice analysis. I want to try to focus on some of the other opportunities that we've identified for leveraging ProteusLib going forward.

So, one avenue that we've started exploring is to use ProteusLib to collaborate with our external partners. As an example of that we've been having early conversations with one of the leading water treatment software companies, OLI Systems specifically, about coupling ProteusLib with their framework. And the motivation for that is we know that the OLI Engine is very good at predicting potential scaling issues in treatment systems, whereas, as Tim just explained, ProteusLib really shines at optimizing complex treatment processes. And so, the idea is that by coupling these two toolsets we should be able to really benefit – or, bring out the benefits of our respective capabilities.

We've also looked into an opportunity to use ProteusLib to study the energy consumption in an actual seawater desalination plants. The plant that we're working with is located in Santa Barbara, California. And what we're trying to do is to understand why this plant is actually consuming more energy than was expected at the design stage. And we're absolutely convinced that ProteusLib can help us understand where we're consuming more energy in the process and just as importantly help us figure out why we're consuming more energy. Our expectation is that we can then generalize those findings and provide general guidance to the community as a whole as to how much energy state-of-the-art desalination facilities should consume.

On a related note, I should also mention that we're working closely with the University of Arizona West where we're using ProteusLib to better understand how we can better operate nanofiltration and reverse osmosis systems.

So, the takeaway message here is that ProteusLib really allows us to engage with our partners in industry and academia. And so, hopefully that's to their benefit but it also gives us a chance to refine our tools as much as possible.

Now, aside from using ProteusLib for collaborations we see opportunities to use it as a foundation to provide new and advanced capabilities. So, let me give you some examples of what we're talking about here. One area that we're excited about is using the framework to improve the way that we design water treatment processes conceptually. And what I'll speak to here is the RO process as one example of that. The current design process for RO systems is fairly iterative in nature. It basically comes down to an experienced designer speculating and specifying how many stages they want to include, how many vessels they're operating in parallel, and how many membrane elements those vessels should include. That one configuration is then typically taken and run through a simulator to check the performance of that design. And then, the designer will go back, make a couple of changes, and repeat the process over and over. So, it's a trial-and-error-based approach, really. And it works, but it's not all that efficient. And what's more is it doesn't really take advantage of modern optimization capabilities.

So, what we're looking to do instead is to take our ProteusLib models and embed them in process design superstructures that consider many, many different possible configurations so that we can then use an optimization routine to extract from that superstructure the one design that is mathematically proven to be optimal for a given objective. And so, we're hoping this will not just speed up the design process but it will also ensure that we're not overlooking potentially promising designs.

Another area that we're very excited about is using ProteusLib to do network-level treatment system analysis. So, looking at a geographically vast area and trying to understand where within that area do we need to place treatment facilities? How big do we make them? And how exactly do we operate them in the context of the network? One area where we think this is very applicable is produced water from oil and gas operations, and I'll talk a bit more about that on the following slide. But the takeaway message here is that ProteusLib, we're hoping, can not only help us find better designs faster, it can also allow us to look at large-scale water treatment processes and find solutions for those.

So, the example that I just touched on is related to produced water from oil and gas operations. And specifically, the opportunity that we want to pursue is to embed ProteusLib models in a network-level modeling framework so that we can understand how we might be able to transform produced water from oil and gas operations from what is currently a waste to a valuable resource.

So, just for context, in oil and gas development areas all across the country we have well sites that are bringing high-TDS water to surface every single day. And what the industry currently does with it is primarily inject that water in underground disposal facilities. To some extent that water is also reused for hydraulic fracking purposes but with very minimal treatment. However, for a number of reasons that I won't get into today we are confident that the industry will start treating and actually desalinating that water in the near future. But that means that companies will have to figure out where within their area do they want to place these small, mobile, modular treatment facilities? How many of them do they need? How much capacity does every one of them have to have? And how exactly – or, which technology exactly do we want to rely on to treat the water? And all of that then allows us to look at how we can reuse treated produced water, whether it's for irrigation or drought protections, or possibly to extract critical materials from the water, such as lithium.

So, the key challenge, though, is that we have to look at these problems from a network-level perspective and consider all of the options that are available to us. And we're confident that ProteusLib can help us build a framework to accomplish just that. So, the takeaway message here is that we're hopeful that we can use this framework to build a toolset that can help us identify produced water management, treatment, and reuse solutions that are not just cost-effective but also environmentally sustainable. So, this is just one example of some of the things that we're excited about in looking to leverage ProteusLib. We're looking forward to sharing more of those examples with everyone in the years to come.

And so, with that I'm going to pass it over to Deb, who is going to talk about ProteusLib accessibility.

>>Deb Agarwal: Thank you so much, Marcus. So, let's go ahead to the next slide. So, first off, I just want to point out at the moment ProteusLib is not yet released. You've seen a couple of questions in the chat related to that. We are hoping to release soon. We're still working with the Department of Energy on the licensing. It is our hope we'll

be able to do a BSD-style license. And the reason why a BSD-style is that gives you broad ability to reuse that code and to modify it and take parts of it and utilize it in other things, including commercial products. The whole idea is to provide this underlying – BSD stands for Berkeley Systems… I actually can't think of the other part. It's – a lot of people think of GNU licenses as the most liberal, but the Berkeley one is actually the most liberal because it does allow commercial reuse, it allows – it's completely open source. Very few clauses to it about what you can and can't do with the code. Basically, the primary clause says that you have to provide attribution of where you got the code.

So, basically, though, the other point that I want to make is that we have a very strong process in place that we put in place for the building of IDAES software and we're bringing into the ProteusLib – the same people are working on it – where we do the professional software development, continuous testing and integration, the documentation, the user engagement, the tutorials, scheduled releases, all of those pieces that you expect in production software. We do all of those things and hopefully you'll see that as soon as we're able to do the release. And there is a GitHub repository. Several people got given the link in the chat. Again, that won't be available until we've gotten permission from DoE for the release. Next slide.

Okay. So, what does that mean for you? Next slide.

So, we want you to get involved in ProteusLib. This is not some deep, dark development that we want to then be pushing out the door at you. We would like people to get involved in the development. So, there's a couple things you can do. You can provide date to help with validation of the models. We've already got a couple of processes that are giving us data, but the more the better.

You can also propose in our FOA process to build ProteusLib models. We'll have the underlying framework. We'd love to have you build models with us of your processes. You could in the FOA process actually propose to build models that incorporate into ProteusLib or that you provide separately.

You can partner with our team to incorporate properties and processes that you'd like to see included in the system. You can volunteer to be part – if you think you're an end user of ProteusLib we'd love to have you volunteer to have our user researchers talk to your team about how you'd like to be able to use it so that we can make sure that the design best fits the way you'd like to use it. We can also help you to participate in stakeholder engagement and things like that that we do. Also, we're planning an upcoming workshop, so watch for the date of announcement. Next slide.

So, these are the GitHub addresses once they're available. And the slides will be posted, so I'm going to go on to the next slide because I want to leave some time for any questions.

And the next two seminars – one's in April. Professor Ilenia – actually, I can't pronounce her name well – of Stanford will be speaking in April. And then in May we'll have Dr.

Aaron Wilson. And so, those are – those should both be – announcements will be coming soon. And we'll be filling out the rest of the year soon as well with speakers. If you – as Steve said, if you'd like to be a speaker please reach out and we'd love to have you.

So, with that I'd like to open it up to questions. I think there's – David, have all the questions in the chat been answered? I noticed there were several coming in.

>>David Miller: Yes, there's one question for Tim regarding the – _____ the results for the salt rejection.

[Crosstalk]

>>Deb Agarwal: And just to –

>>Tim Bartholomew: Yeah, I can see it.

>>Deb Agarwal: Before Tim answers that, just a quick reminder. If you'd like to ask a question on audio just raise your hand and you'll be unmuted.

>>Tim Bartholomew: Sure. Yeah. In audio it's much easier to answer the questions too. I was writing up a couple sentences. But anyways, Zahid, there is a big difference between our preliminary results – these are just our initial models; they're not finalized – and what the Wang paper did earlier last year. They only did theoretical energy consumption. They did not calculate water flux or membrane area. So, they weren't calculating cost or things like that. And they weren't including pump efficiency losses and pressure drops, which our preliminary models are doing. So, we had significantly higher energy consumption and the process was cost-optimized, so it was balancing operating and capital cost. So, it was pretty different, but yes. That's the answer to that.

>>David Miller: Okay.

>>Deb Agarwal: All right. Any other questions?

>>David Miller: Nope. That question got answered, so…

>>Tim Bartholomew: There was some questions from Peter Veiss, I think. I don't know if they got answered by Marcus. Oh, it did. Okay. Yeah, we are going to use time-dependent supply chain optimization for water management. That's really, really important for produced water because it varies so much over the years.

>>Deb Agarwal: Okay. So, I want to thank everybody for participating in the webinar. Hopefully you now know a lot more about ProteusLib and some of our plans for that. Hopefully that's also motivated you to get involved. Feel free to contact us to get involved. We'd love to have you be building models, be joining in to provide data, and also signing up as a potential future user of the system. We want all of the above. So, please feel free to contact Steve or I or David or any of the team to sign up. And with that

we want to thank you all for your participation and hope to see you again soon. Thank you.

[End of Audio]