RESOURCES

Home
Resources
Resource Center

OPUS Workflows

Reference Link: https://www.tracelink.com/resources/tracelink-university/opus-solution-designer-foundations-presented-futurelink

LJ Hall: Hi. My name is LJ Hall, and I'll speak to you today about OPUS workflows on the platform. I wanted to give a bit of background about me first. I've spent over 11 years now in the life sciences industry. I started by working on a mobile solution for pharma supply chain management at a company called Rock I.T. Solutions, and we were acquired by TraceLink in 2017.

Over those eleven and a half years, I've done three years in QA, three years in tech consulting at both Rock I.T. Solutions and TraceLink, and then six years of product management, with the last four specifically being on the OPUS platform.

I've seen the life sciences industry from a lot of different perspectives. The reason I'm sharing this with you today is that I can tell from firsthand experience in both tech consulting roles and QA roles how difficult it can be in the old world to configure software for your businesses with commercial off-the-shelf products.

Which is why I'm excited to be up here today to talk about what we can do with the new world, building off of what Jerry presented, and what Siva presented, along with what Bob presented this morning.

Workflows on the OPUS platform are an important concept to understand in order to maximize the value you can achieve for your business. I'm going to guide you through what workflows are, how users are going to interact with them, how to configure them in OPUS solution environment, and how your businesses can use them to your benefit.

Workflows don't sit in the same plane of existence as business objects that Jerry and Siva were showing us. An end-user can't go create an instance of a workflow. Instead, they're associated with business objects and help drive the structure and experience of that object for the end-users.

It also helps the user stay within the lines of what the OPUS developer has intended. It may allow or prevent a field from being modified until after you reach a certain state in the workflow. They also help drive a consistent experience across objects. This is something that Siva and Jerry touched upon as well. We want to give that consistent experience.

Once end users are familiar with how an object with a workflow will behave and what that pattern looks like on OPUS, they'll be able to perform new business processes in your organization more efficiently and quickly.

On the OPUS platform, all objects with workflows will be displayed in this workflow picker. We saw this a bit in Siva's demo as well. We'll get into the structure of that specifically later on. Everything I listed there are some of the values that you're going to get just with out-of-the-box workflows.

Then OSE also provides an interface to configure workflows, and so we're going to go explore that later on as well with the demo.

I said that workflows are associated to business objects. This decision and association is made by OPUS developers, and therefore it's going to be predefined for us out-of-the-box, and we call these standard workflows.

Standard workflow will be a sequence of states and transitions. The OPUS developer will have tied necessary app logic to those states and transitions, and you can always count on these happening.

If you choose to configure a solution in the way that we saw with Siva's demo, the standard workflows become company workflows inside the new solution. This is where we can extend the functionality of those standard workflows to handle our specific business processes.

We're going to add substates, which are associated to a base state. We can add transitions from and to our substates. In this example here, we have the standard workflow. If we choose to configure, we can add substates to the base state, and then transitions to and from the base states and to each other. This is where we can inject our own logic into what we need to do for our businesses.

I'm not going to cover every single rule that we have today, but one thing I do want to point out is for our transitions that we add, it has to form an honored pair with the base date transitions. For example, I cannot create a transition from drafting back to to-do.

The reason for that is quite simple. The app developers have defined business logic that they want tied to those transitions and so we can't violate that. Beyond that, we're going to talk about transition conditions and post-transition actions in a little bit. This is where we're going to see a lot of that added logic that we want for our business.

One axis is the standard workflows and the company workflows. Another that I want to talk about is the different classes of objects that we're going to see with workflows. One are business objects, and then the other are business transaction objects.

A business object is going to be the simpler objects, something like a task or an issue in a ticket management system. The other is a business transaction object. The workflows for these tend to be a bit more complex, and they are able to accomplish a lot more.

Really, the defining feature of business transaction objects is the workflow is going to be run automatically, and that gives us a lot of power for what we need to do.

They will be run without intervention, but if they need to, they can pause. They can wait for user intervention. You can get a notification, an email about the transaction being paused. So, our B2B transactions are going to come with this class out-of-the-box for you.

Then since these are all modeled with the same metadata, we get to tie into the same user interface and that same consistent experience that Siva was able to demo for us, meaning that both your business object and business transaction objects will be displayed in the same way.

For the business transaction objects, this is quite powerful because if, for whatever reason, it pauses or it gets stuck in the middle, you could always see how it's persisted and you can always view that object.

On the configuration side, all standard workflows come with base states and base state transitions. These are defined by the OPUS developer and they define the life cycle of an object type.

When we configure company workflows, we're going to add substates to our base states, which can provide additional insight into the context of our business. Then from there, we're going to add transitions to and from our substates and to and from other base states as well to our substates.

I'll also mention here that the same concept applies where if you perform a transition, the same app logic is going to be performed from the base state transition. In this example, if you transition from in review to done, it's also going to transition from in progress to done and trigger the logic that the app developer defined for that.

Now, down into the transition conditions and post-transition actions, these are quite powerful. I think that these are self-descriptive in what they do. Transition conditions, they are logical conditions that are tied to a transition, and define whether or not the transition can be performed.

For example, if a particular attribute is empty or greater than a certain value, then the check could fail and it could block the transition for the user.

Then post-transition actions are actions that you configure to happen after a transition occurs. You could send a notification. You can send an email, send another B2B message, send something to a backend system. This is where we can get a lot of power from the configurability.

Then what I do want to mention is right now these are written in JavaScript, but we're coming with more tools available to build these from the no-code interface as well. We're really excited about this feature and where this is at.

Movie time, we're going to go back to Sally Solutions and Sierra Biowell from the previous example. We're going to take the purchase order object that Siva demoed, and we're going to add our own configured workflow to it. We're going to add two substates.

In the demo, we're just going to add transitions to Submitted from both draft PO and pending approval, which are our two substates, and then a transition between them. And we're going to have a transition condition between the two substates, and we're going to see how that looks like in action.

The use case here is we want there to be a pending approval state between moving from draft to Submitted, and this is going to allow Sierra Biowell to approve purchases before they go through.

The first scene here, we're going to watch this with the base standard workflow. What we're going to see is at the top of the screen, when it loads, we see that the first state there was draft. We're going to enter in some information and save. We're going to go back in, we see that it's draft. Let's select move to, we're going to move to submit from draft. Save that.

Scene two, now let's log in with Sally Solutions. She's going to log into the OPUS Solution environment. Here's the search workflows page within our company solution. We're going to search for the solution that we want to configure.

[pause]

LJ: We're going to pick our company workflow. As you can see, there's our base states. Let's go ahead and add a substate. So we're going to associate this to the draft state. We're going to call it draft PO.

Let's add another one. We're going to add pending approval. It's also going to be a subset of draft.

[pause]

LJ: Now we have our substates, now we're going to add transitions between them. What we're going to do for this is we're going to pick the substrate from the table here. You'll see that we'll be able to open the, click on the substrate row.

We have an icon for adding a transition, so let's add a transition to Submitted. Now we're also going to add a transition to draft PO. This is the JavaScript for the transition condition. It's going to go from draft PO to pending approval. We're going to see that in action once we experience this again with the configured company workflow.

Then finally, we'll add a transition from pending approval to Submitted. We'll be able to see this all play out when we go back into a new instance of the object from Sierra Biowell's perspective.

We have our substates. We can see the transitions from them. Now we go back in, we create a new purchase order. As you can see, the substate is displayed here. Again, no reloading, it just happens after you save the workflow. So we're going to create this new purchase order. It's in a draft PO state. From here, let's go into edit. Enter in some information.

Now we're going to try the transition. The transition condition was looking at one of the attributes to see if it had a value checked. If we go in here, let's now experience what it looks like when we go to a substate. Now this move-to becomes a dropdown.

We can try to move to pending approval, and the first time we're going to try this, we're not going to enter in the action field, so we click save. We're going to get an error here, we're blocked from transitioning because the transition condition is making the check for us that we wanted. Now we can try it again. This time we're going to select in action, which is the field we're looking at.

[pause]

LJ: That's it. From here, we move into the pending approval state like we wanted. From here, whoever the approver is, they can go back in, they can edit it, and they will be able to move it to Submitted.

[pause]

LJ: We just experienced the three scenes. The first one was the standard workflow, no customization, no configuration. The second one was configuring the company workflow, and then the third one was experiencing the configured company workflow.

Some key takeaways from workflows, workflows offer streamlined process management. Workflows facilitate the structured movement of business objects through states and transitions, ensure efficient process management. They offer efficiency and flexibility.

Workflows enhance business performance by automating processes, ensuring consistency, and allowing configuration to meet specific needs. They're both ready out-of-the-box and configurable. Standard workflows provide immediate functionality, while company workflows offer fine-grain control with substates, transition conditions, and post-transition actions.

We can provide adaptable workflow design. Solution designers can configure workflows for both simple and complex business processes, ensuring alignment with unique operational needs.

Workflows support integration and specificity. So, workflows can integrate with our other processes and include specific conditions and transitions, which give us tight control and seamless data flow across our business functions.