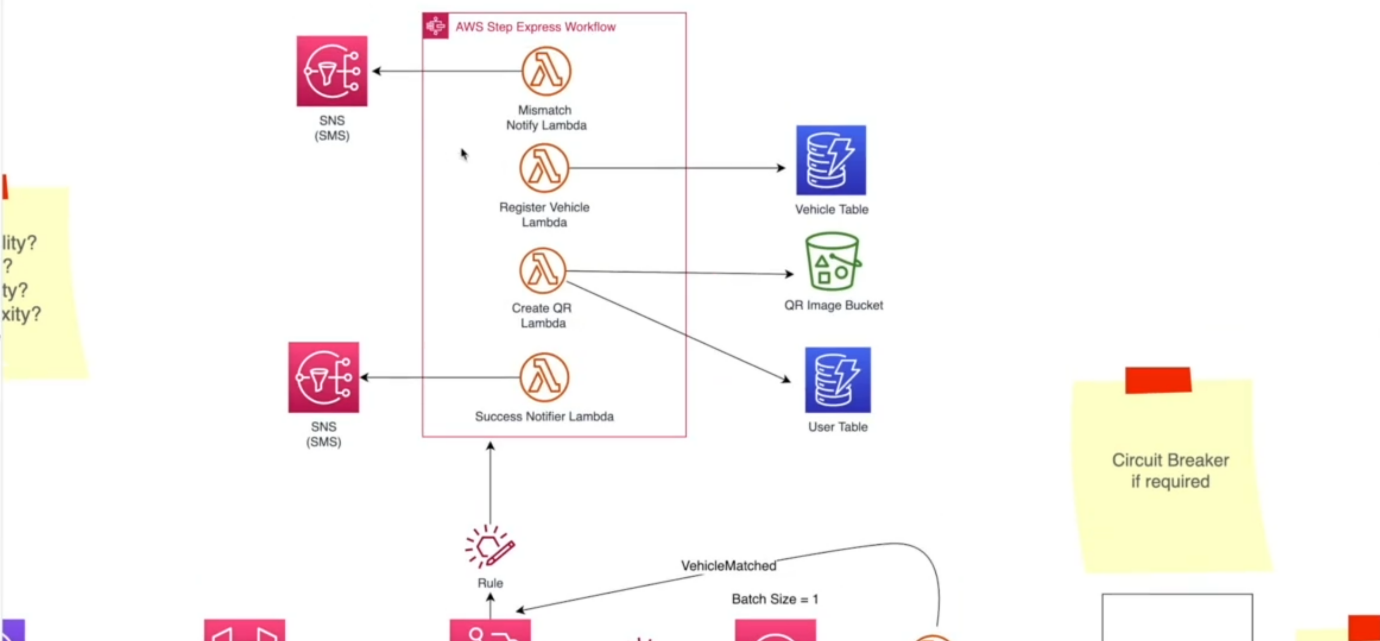
**11. Managing Serverless Workflows with State Machines**



* let us talk about this section of the architecture, you will get a message from the external system whether the information is valid or invalid and there will be a role listening into that type of events. It will invoke a step function express workflow.

--- **So, what really is a step function?**

* So, its step functions are a managed service from AWS that will help you to manage your workflows. So, in our case, our workflow would be whenever we receive some confirmation from the external system, we should check whether the information is valid. If they are not valid, they must notify the users. The detail that you have entered is invalid. Please try again.
* If that is valid, we will skip this step and we will get to the register vehicle step. In this case, we will add the registration information into maybe a vehicle table. So, this is a Dynamo DB table.
* So here we will store the vehicle number, vehicle type and these details in this table and after that we need to generate a QR code for the user and this lambda function will take care of it. It upgrades the QR code and it will also create the QR code image and add it to an S3 bucket and we will call this s3 bucket QR image bucket and after that the same lambda function will get the history URL of that QR code and add that URL to the user table.
* So, our user table we will include user details, first name, last name, address as well as QR code image link and that will store a link to the S3 image.
* So, as you can see, the set of things that we need to execute in a certain sequence now we can use a step function to orchestrate these workflows. So as part of the orchestration, it will invoke the correct lambda function by looking at the input data information.
* In this case, from our event rule, we will receive whether the user information and the vehicle information are matching or not. So, depending upon that boolean value, we will invoke this lambda function.

--- **what if this lambda fails to generate a QR code?**

* Then we can create another failure step and send an SMS to the user. The vehicle registration process failed due to this reason, but however, this earliest step has already been executed, the user's data or the vehicle data is already added to the vehicle table. So, if we are to completely reset these states. We should remove that information and AWS step function make that process easy.
* We can use patterns like saga patterns that we can instruct the step function to clear out the previous states. So that will also be handled by a step function as a orchestration service.
* So, if there is any failure for a particular user, we can get that request correlation ID and look at this workflow and, in that workflow, we will see, all right, something has gone wrong with in creating the QR code and then we can further observe it.
* What input has it received from the previous step and what was the error? All these things are visible to you. So, error handling is easy when we have a workflow orchestration service like step function added to our application.

**Now one might be wondering, are we able to use this event bridge again to handle these steps?**

* For example, when we get this vehicle matched, event added to the event bridge, we will create another rule and we can invoke this lambda function in the event rule, we can particularly say check if the request is failed. If so, invoke this lambda function.
* We can create another rule. That rule will check whether the request from the RMV system is successful. If so, we can send that event to this vehicle registration lambda and once the vehicle registration Lambda added that information to the vehicle table that Lambda can send another event to the event bridge vehicle data saved and again, we can have another rule listening to that event and all this matching event will be sent to this create QR code Lambda.
* We can use event bridge and we can completely take out this step function workflow orchestration, part of it and of course we can do that. then our system will be quite decoupled and that would be a good thing but however, there is a cost to it.
* The cost is mainly about troubleshooting for errors, as well as this application getting very complex.
* If there is a newcomer to your team, he will have a very hard time understanding these flaws but when we have a step function express workflow, we can see how one step calling the other step, what are the retrace steps as well, and how the errors are being handled and all this information is graphically available for him or her and the troubleshooting is also quite easy.

--- **standard workflow**

* We must use step function standard workflow because standard workflow supports asynchronous requests but I would avoid using standard workflow here because standard workflows are expensive in standard workflow. You are being charged for the state transitions. One of the state transitions would be registering your vehicle information and after that calling this QR code lambda and then send the success notifier and so on.
* you will be charged $25 per million state transitions but one of these executions will contain many state transitions. So ultimately, when you run your application in scale, you will probably see the standard workflows are expensive

--- **express workflow**

* on the other hand, express workflows do not charge you by the state transitions. It will charge you by the execution time the same pricing model that you have with Lambda and on the other hand, we can invoke many express workflows in parallel It is highly scalable. I think we can invoke about 100,000 express workflows in parallel but the limitations are we cannot use this type of asynchronous workflows attached to our workflow as well as there is a five-minute duration limit.
* So that is why we handle our asynchronous flow through event bridge, directly calling SQS and the rest of the steps through express workflow Because most of these events are synchronous because these events are synchronous. Registering a vehicle in vehicle table, adding the QR code image into S3 and saving that S3 URL in user table are synchronous and dynamo DB S3 and this can absorb a lot of traffic, so we can call them synchronously.

--- **the orchestration patterns**.

* if you are to use, let us imagine, event bridge to handle all these events in a very loosely coupled manner using rules and so on, and that is called choreography pattern. So here we use both choreography pattern as well as orchestration pattern.
* The orchestration pattern is express workflow.
* So, at the end of our workflow, the users will receive an SMS that their QR code is ready with a link to the login page.