Upgrade your NiFi process groups

NiFi has a nice registry function to manage versioning of process groups it is called nifi-registry <https://nifi.apache.org/registry.html>

In this article I will show how you can maintain versions with nifi-registry and how you can upgrade or downgrade all your instances of your process groups.

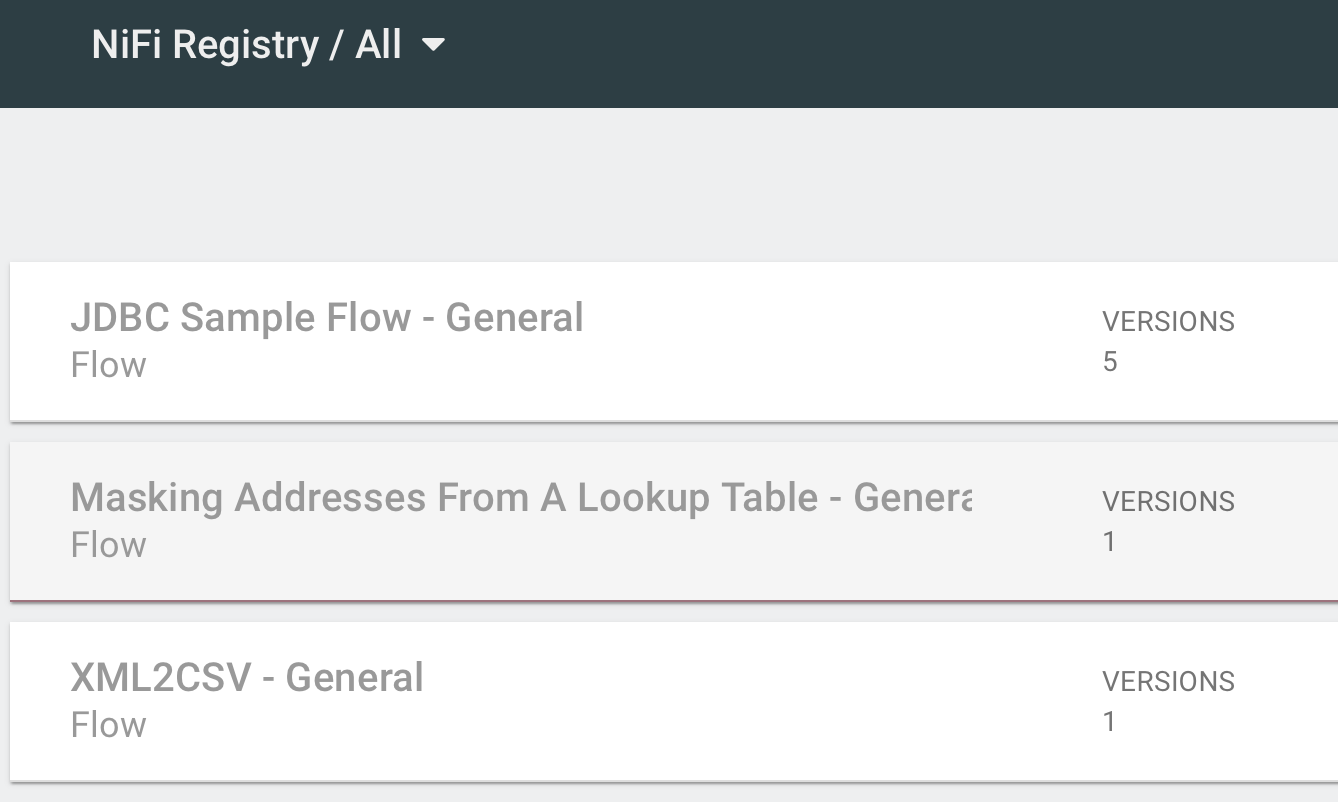
I will not cover how you get started with nifi-registry as it is already well documented in this video [Getting Started With Apache NiFi Registry](https://nifi.apache.org/registry.html).

I will instead show you my nifi-registry and my nifi canvas to describe how you can work with multiple instances of a process group and version control it and put it into production which means upgrading all instances to the version you like to use.

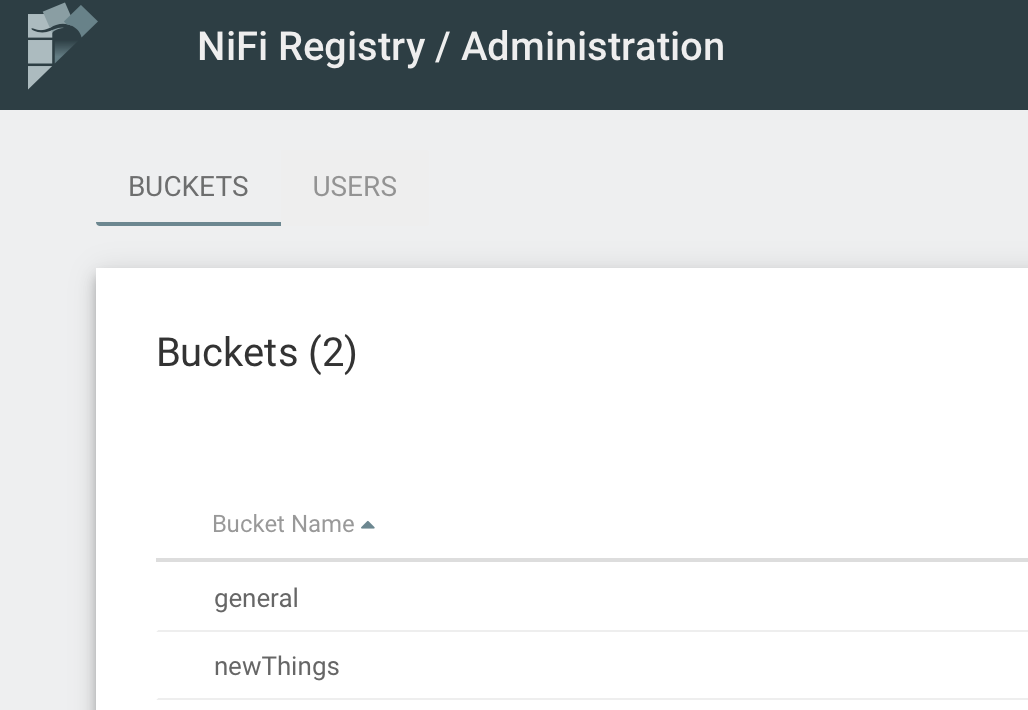
The code is written in groovy and requires following two jar file that you can collect from maven repository httpclient-4.5.5.jar & httpcore-4.4.10.jar.

Source code is available in my git repository <https://github.com/maxbback/nifi-registry-upgrade>

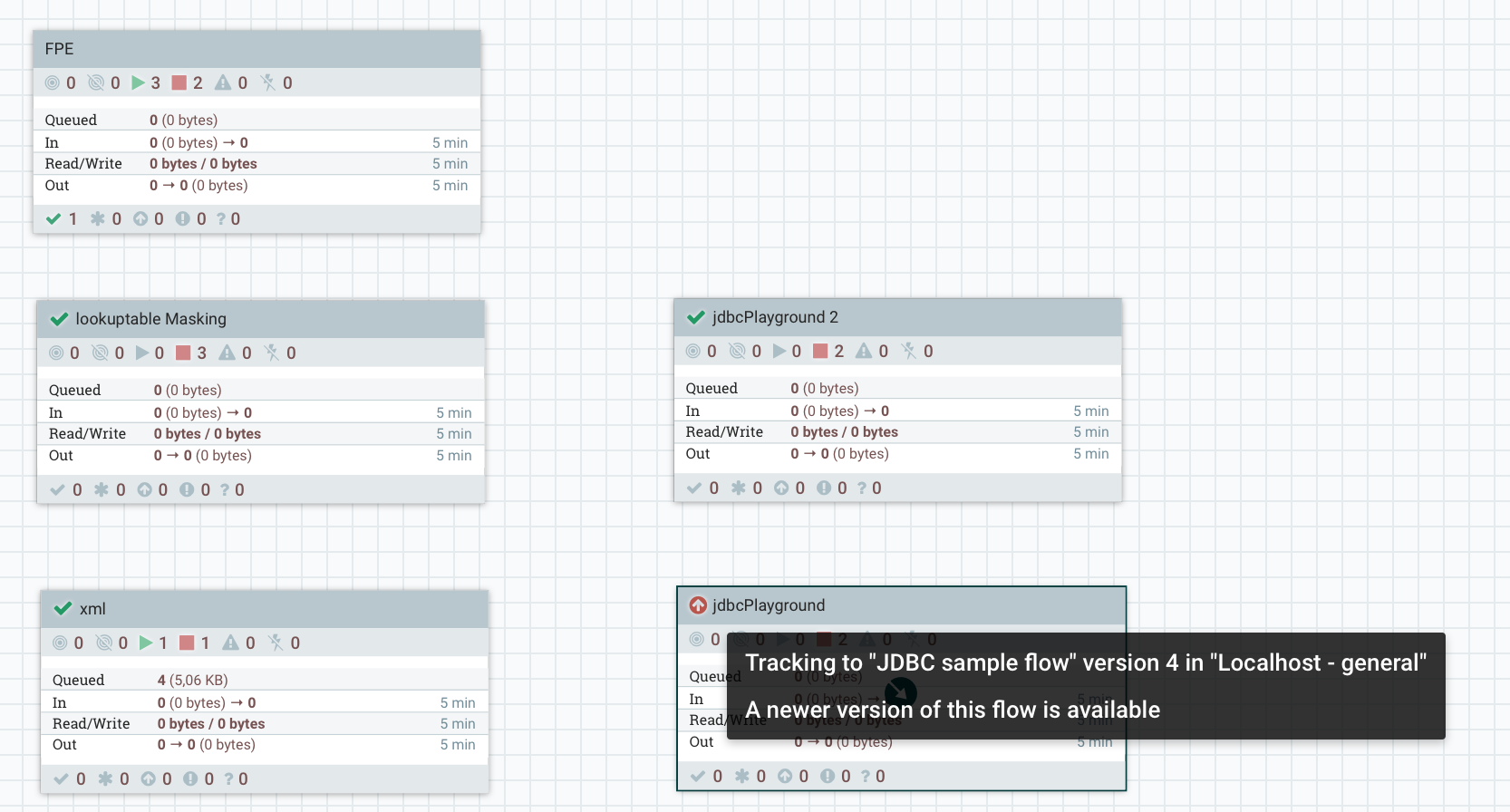
In my nifi-registry I have 3 versioned flows in my bucket “General” and you can see that the flow “JDBC Samle Flow” has 5 versions



A Bucket is a collection of flows, and you can use this to categorize your flows.



In NiFi canvas we can see the process groups I have



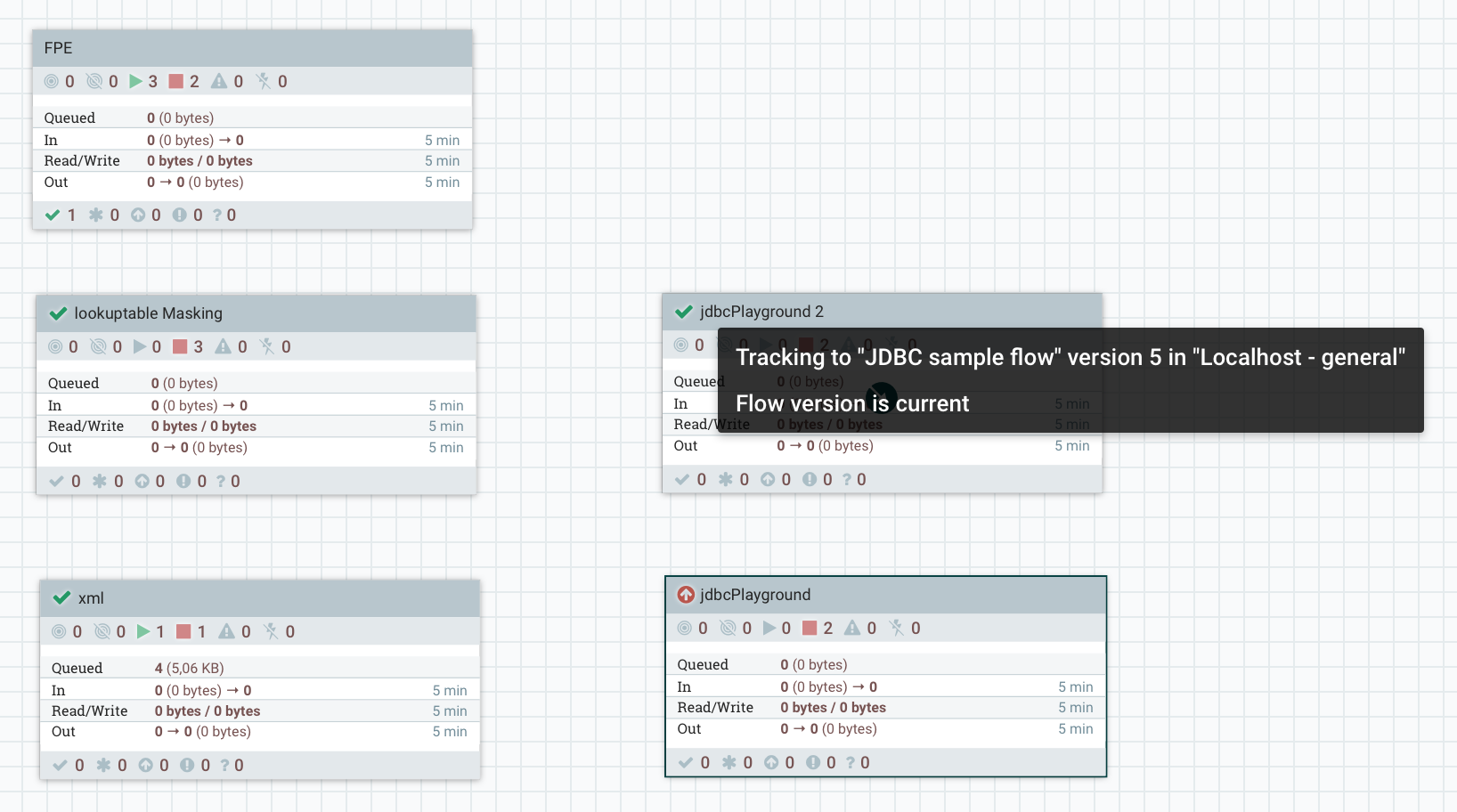
If the process group has an icon in the left corner it indicates that the process group is under version control.

If we hover the mouse over the icon a popup window is visible with information of the version and the name of the version flow.

In above picture we can see that processgroup jdbCPlayground is of version 4 and that the flow name in nifi-registry is “JDBC sample flow”.

We can also see that the icon in the left corner is a red cyrcle with an arrow in it, wich indicates it is not using the latest version.

If we look at process group jdbcPlayground 2 we can see that the icon is a green confirm icon wich indicates that we use latest version, and if we hover over it we can see the flow name is “JDBC sample flow” the same flow name as was used for process group jdbcPlayground.



From this we learn that we can get information of version and the nifi-registry flow name from hovering over the version icon, we also learn that process group name do not have to match the name of the nifi-registry flow name and that we can have any number of instances of a nifi-registry flow.

Let’s assume that we have finished development and testing of our new version of “JDBC sample flow” and committed it as the version 5 with the help of the process group instance “jdbcPlayground 2” and we now want to upgrade all process group instances that is using the version flow “JDBC sample flow” which we have 2 instances of in this nifi canvas.

In the groovy code we define that we shall use latest version and the flow name of the process groups to upgrade like this

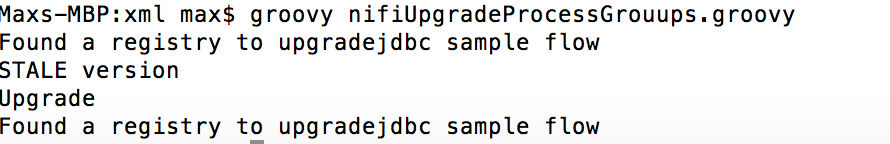
// Nifi-registry flow file name

registryFlowName = "JDBC sample flow"

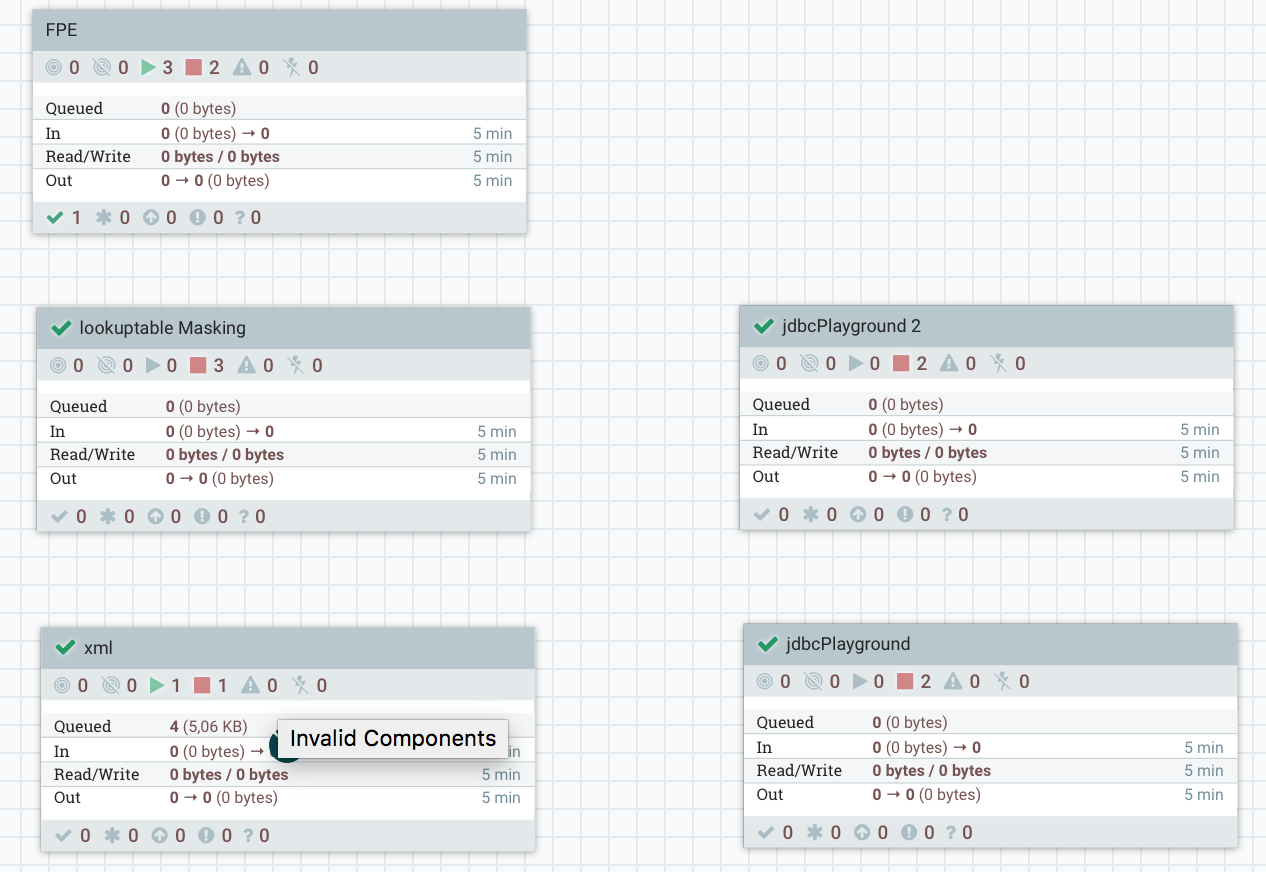
// target version for upgrade if 0 upgrade to latest version

targetVersion = 0

And now we run it;



We can see it found one instance to upgrade, and if we look at the canvas now it looks like this.



And we can see that process group jdbcPlayground has a green checkbox in the corner which indicates it is up to latest version.

Let’s assume we found a problem and needs to downgrade to version 4, we change the parameters of the version in the script;

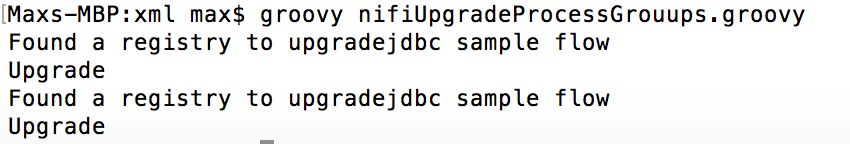
// Nifi-registry flow file name

registryFlowName = "JDBC sample flow"

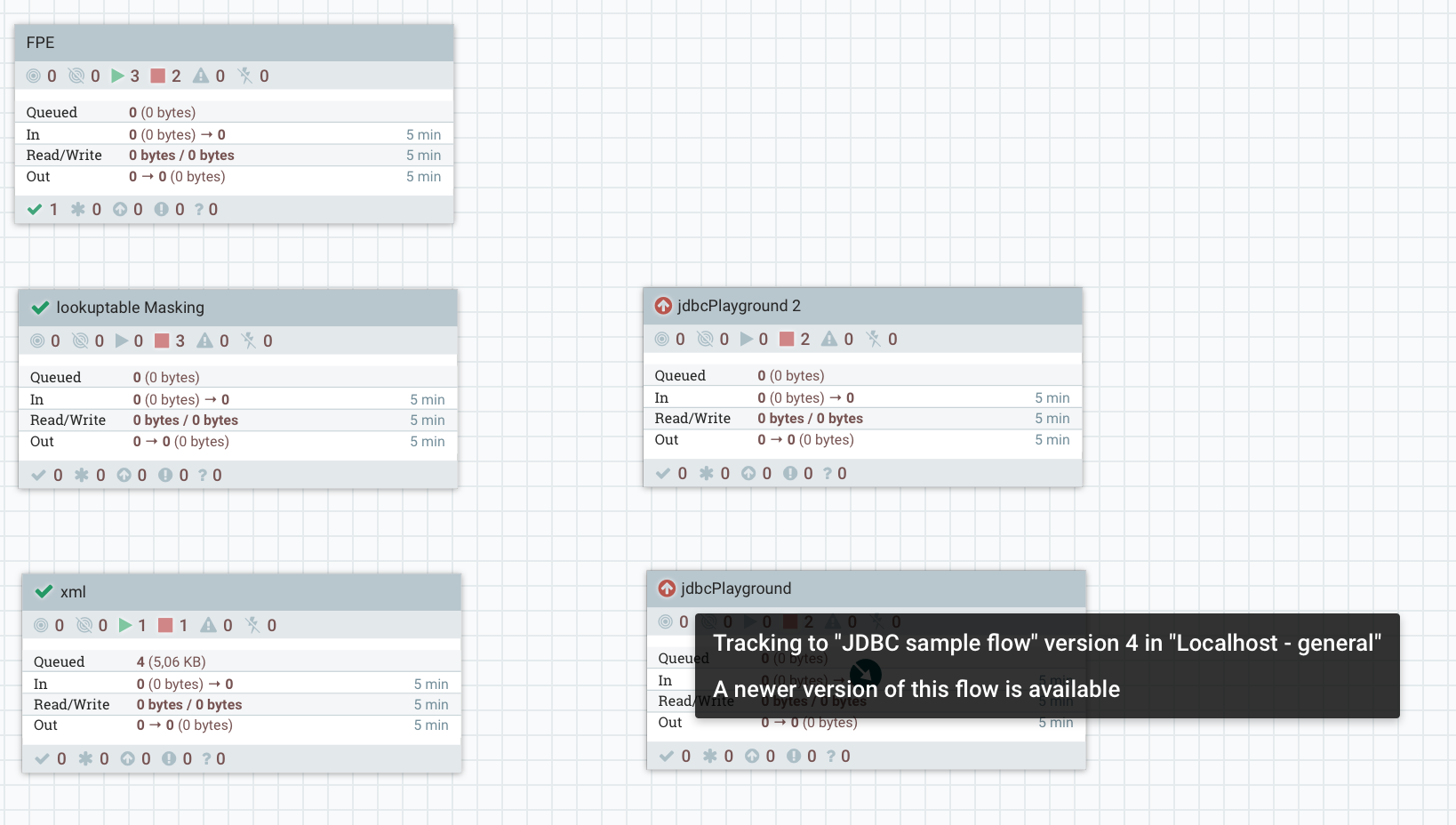
// target version for upgrade if 0 upgrade to latest version

targetVersion = 4

And run the script again;



We can see that it found 2 instances to upgrade or really downgrade to version 4, lets check NiFi canvas.



We can now see that both process groups has a red circle indicated that it needs to be upgraded, and if we hover on it we can see it is version 4 that is deployed.

You can use this script as part of your deployment pipeline to secure that all process groups are upgraded to the same version and if needed as we did here roll back to previous version.