

RADS

Version Control

<https://github.com/radservice/rad-community/fork>

Prerequisites

1. Good understanding on concept of Application in RADS with the know-hows to create Process, Form, List, and Userview.
2. Basic understanding on versioning.

Content

1. Introduction to Version Control
2. Process Version Control
3. Application Version Control
4. Git Version Control



Chapter I

Introduction to Version Control

Version Control

- There are 3 types of version controls available in RADS App management. They are:-
 - Application Version.
 - Process Version.
 - Git Version.

Version Control

- Which, when, and how do we make use of version control?
 - Process Design Fixes or Update.
 - Application Form / List / Userview design update.

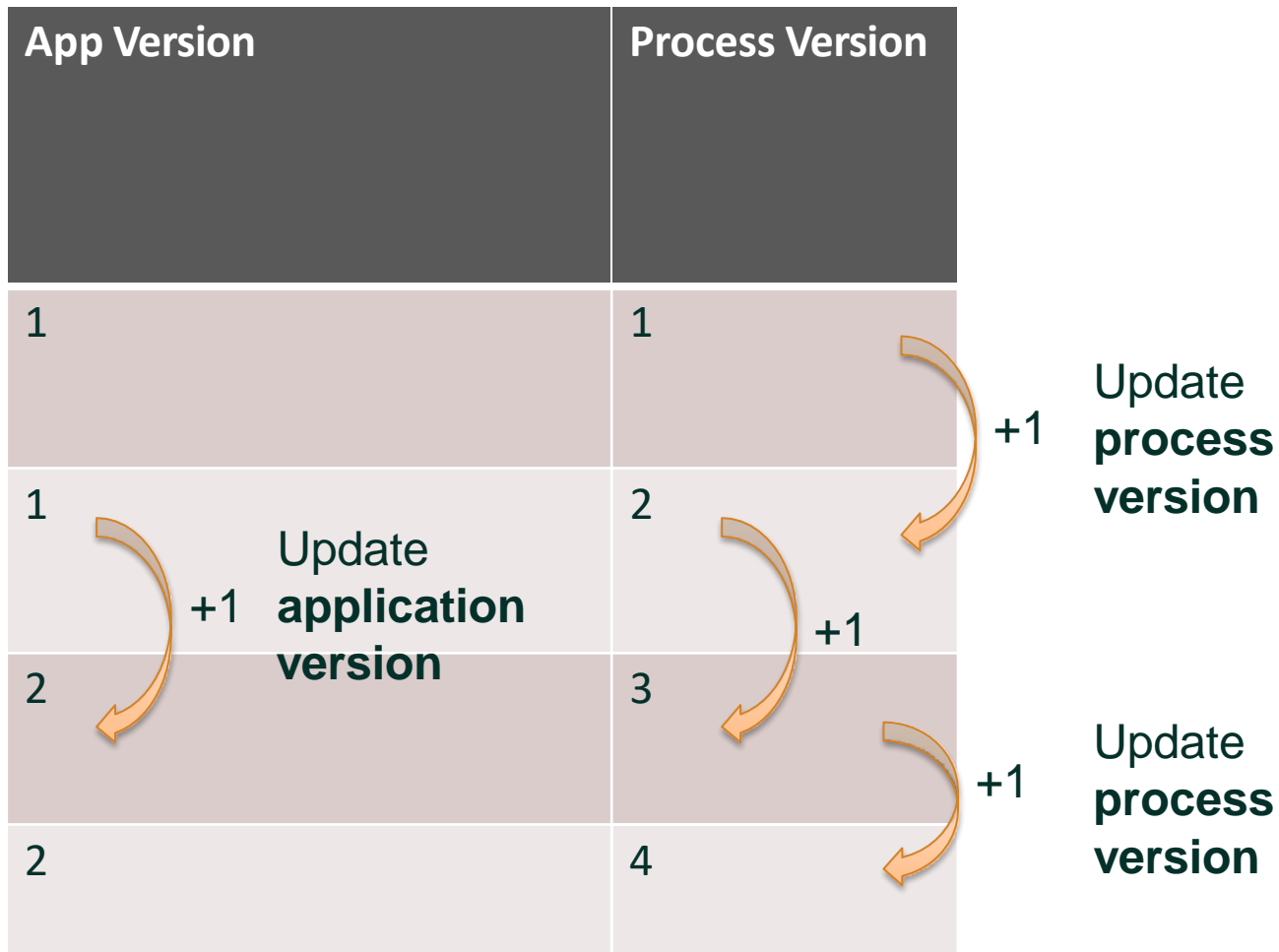
Version Control

- Updating Process version
 - Updates only the Processes under the current Application version.
 - **Updates** existing running instances of the processes found under the current Application version to the new process design.
- Updating Application version
 - Makes a copy of the Processes, Forms, Lists, Userviews of the current version to the new version.
 - Includes all the Processes, Forms, Datalists, and Userviews.
 - Does **NOT** affect any running process instances.

Version Control

Action / Components	Update Process Version	Update Application Version
Process	✓	✓
Form		✓
List		✓
Userview		✓
Application Settings		✓

Version Control



Version Control

App Version	Process Version	Migrate existing running instances of the current App version to new Process version
1	1	
1	2	<ul style="list-style-type: none">• Yes (All that are created under current App version)
2	3	<ul style="list-style-type: none">• No
2	4	<ul style="list-style-type: none">• Yes (All that are created under current App version)• Will not affect instances of App version 1)



Update
**process
version**



Update
**application
version**



Update
**process
version**

Use Cases

- Updating the Process version is ideal when:-
 1. Urgent update to process design flaw.
- Updating the Application version is ideal when:-
 1. Application is ready to be pushed to production.
 2. Completed design ready to be backed up as a version/backup before moving on to the next iteration of development.

Chapter Review

- Understand the various types of version control.



Chapter 2

Process Version Control

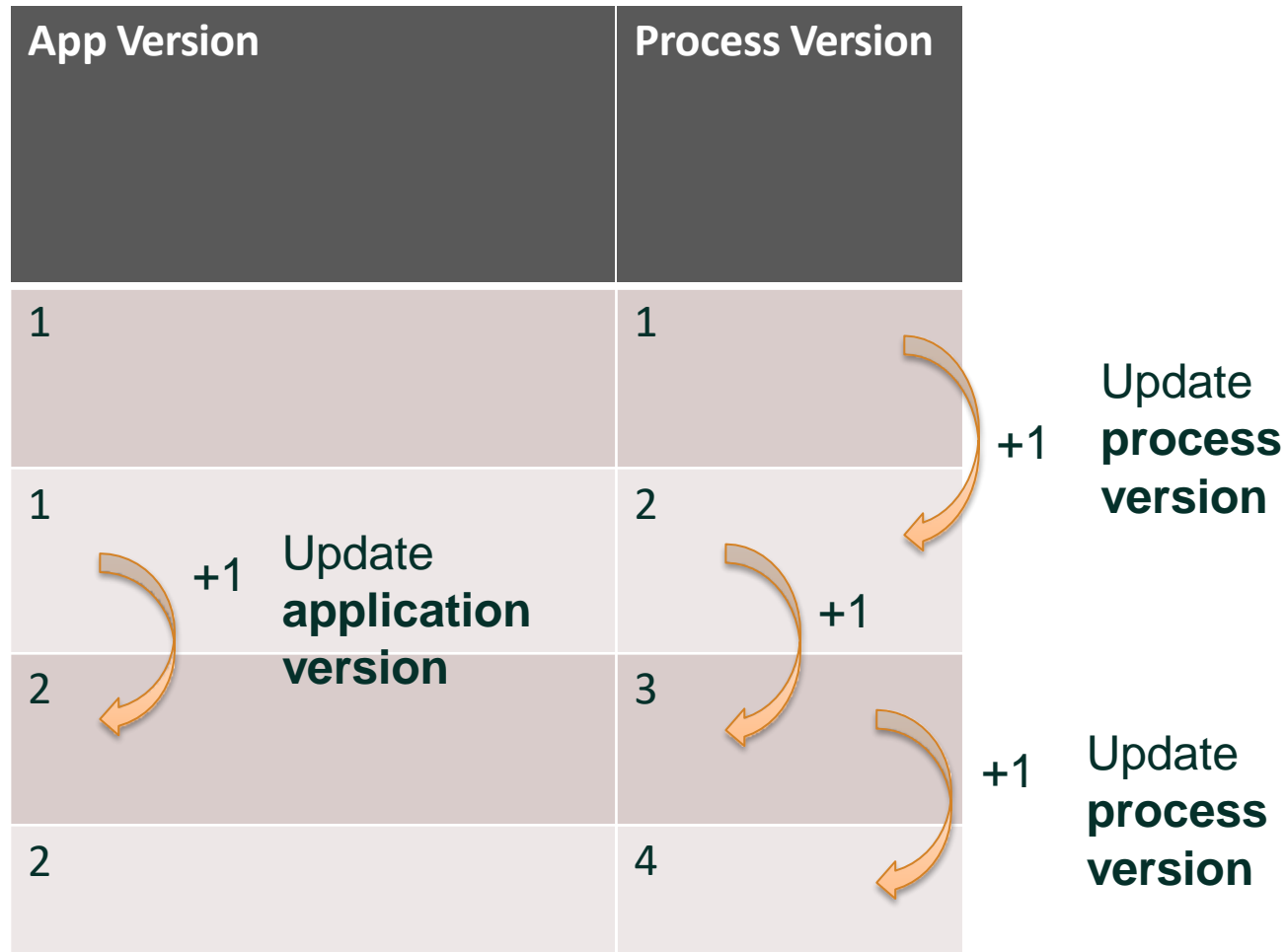
Process Version

- There may be multiple process versions tagged to one Application version.
- However, there can be only one active Process Version (the latest) in an Application version at any point of time.
- It is NOT possible to rollback to earlier process version in the same application version.

Version Control

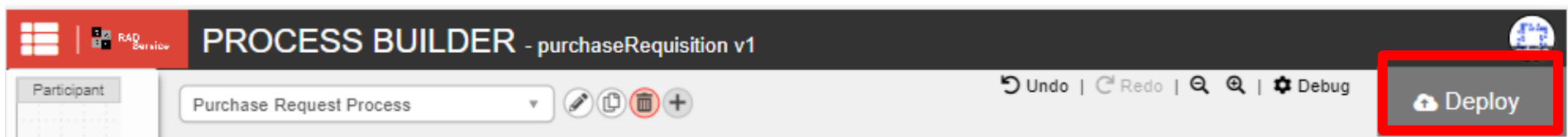
Action / Components	Update Process Version	Update Application Version
Process	✓	✓
Form		✓
List		✓
Userview		✓
Application Settings		✓

Version Control

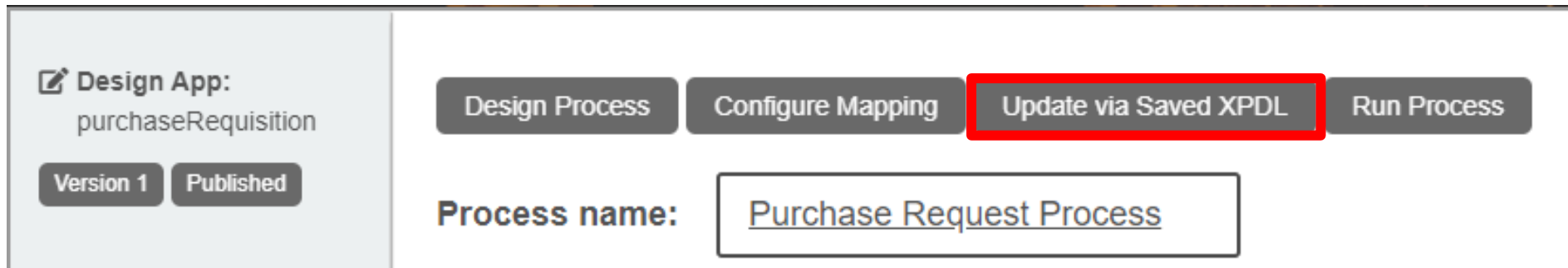


How to Update Process Version?

- Upon deployment from the Workflow Designer.

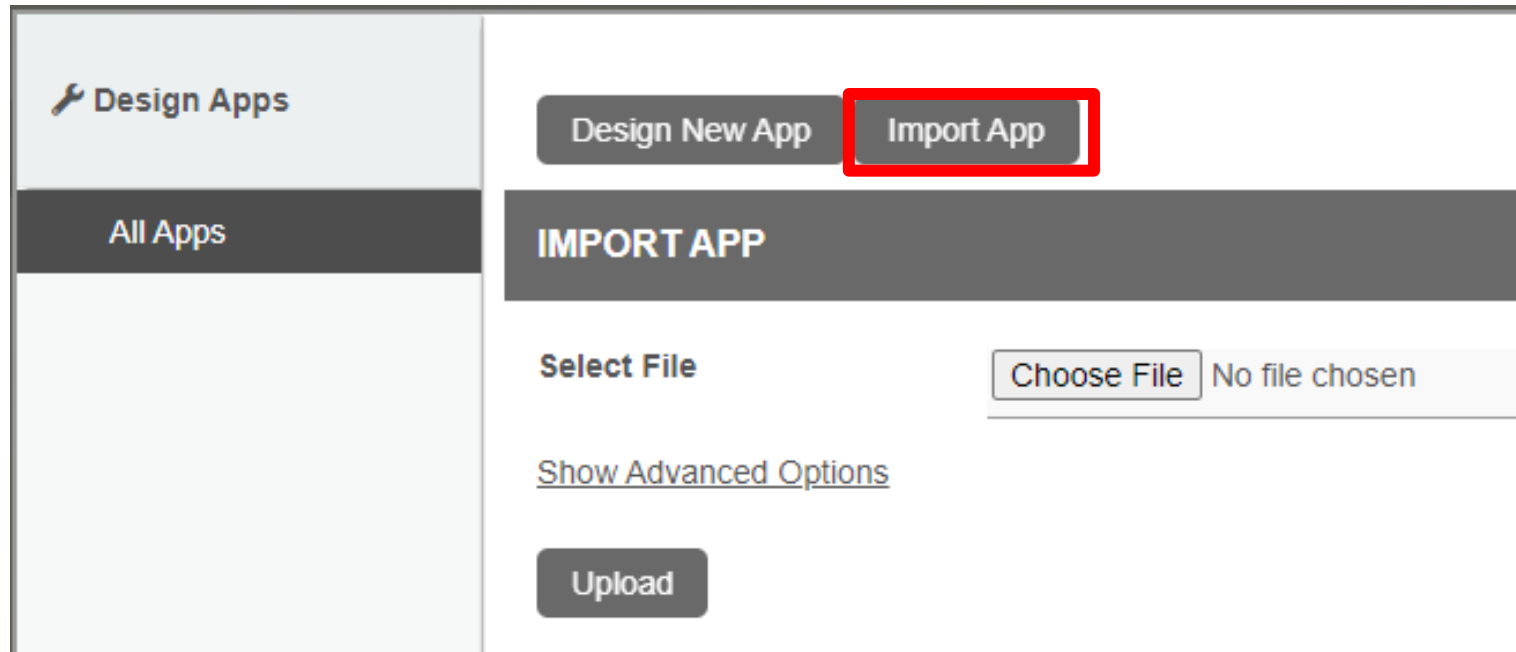


- Update via Saved XPDL from the application designer.



How to Update Process Version?

- By updating App version - Upon import of App (of the same App ID)



(This will increase App Version too, more on this later)

Migration of Process Instances

- On the event of process update, process instances that are still running on the current process version will be migrated/updated to the latest process version (in the same app version only).

Migration of Process Instances - Missing Activity

- If there's NO matching activity(ies) that can be matched, RADS will **NOT migrate the process instance** and *it get **aborted***.
- Important Notes as opposed to Joget Workflow v6:
 - Staying in its original process instance is a new behavior in RADS. In Joget Workflow v6, the original process instance will be aborted, and a new process instance will be created, resuming where it was last left off.
 - As it stays in its original instance, SLA and relevant attribute data are kept intact, instead of getting resetted.
- More reading at:
<https://docs.rads.purwana.net/Update+Existing+Running+Process+Instances+to+the+Newer+Process+Flow+After+Process+Changes>

Important Note

- The newly created activity instances **will continue to function as if nothing has changed** and should be transparent to the end users.
- Resumed activities will continue to use previously mapped forms.

Map Participants to Users

Map Activities to Forms

This is the list of activities defined in the Workflow design.

Submit Request
ID : submitRequest

Form Name

1 - Submit Request

Remove Mapping

☐ Remove Save as Draft Button

☐ Show The Next Assignment When Completed

Important Note

- If there's new activity(ies)/tool(s) being added to the new process design. One shall need to configure the mapping(s) accordingly.

Chapter Review

- Understand on how Process Version works.
- Understand its implications on existing running process instances.



Chapter 3

Application Version Control

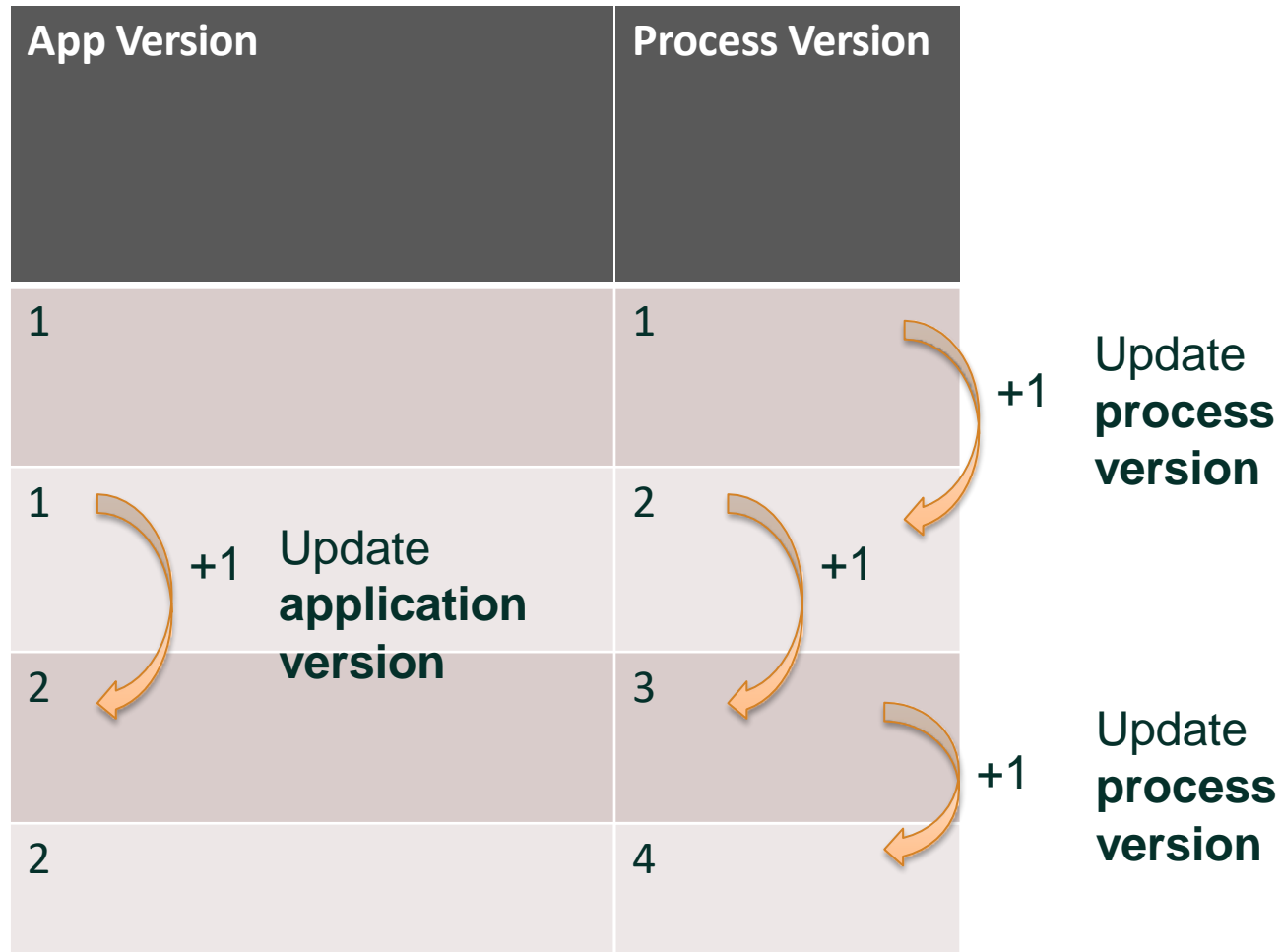
Application Version

- Application version consists of the following:-
 - Processes
 - Forms
 - Lists
 - Userviews
- Each Application version would contain only one Process version (the latest) at any point of time.

Version Control

Action / Components	Update Process Version	Update Application Version
Process	✓	✓
Form		✓
List		✓
Userview		✓
Application Settings		✓

Version Control



How To Update Application Version?

1. App Control Panel > Versions > Select version > New Version

APP CONFIGURATION MANAGEMENT

Manage App Version Delegate App Designer Role

<input type="radio"/>	VERSION	PUBLISHED	NOTES	DATE
<input type="radio"/>	1	✓		18-10- AM

15 Page 1 of 1 Displaying 1 to 1 of 1 items

New Version Publish Unpublish View Delete

By creating a **New Version**, the App design will be **cloned** into the new version.

Online Reference:

<https://docs.rads.purwana.net/App+Versioning+and+Publishing>

How To Update Application Version?

2. Import App

- By importing the app into a RADS server, the Application Version will **increase by 1** over the existing version already in the server.

What does this means?

When you are dealing with the same app across different RADS servers, you may end up with different app version in each server but with exact same app design.

App Version Across Different Servers

- When you are dealing with the same app across different RADS servers, you may end up with different app version in each server but with exact same app design.

Development Server

Design App: Purchase Requisition

Version 8

Forms & UI

Processes

Properties & Export

Performance

Logs

Notes Environment Variable Message Re

version 3 2019102
- Added deadline to escalate and send reminder

version 2 20191018
- Added email tool
- Added datalist and userview for end user to access

version 1 20191015
- Created form and process for purchase requisition

Production Server

Design App: Purchase Requisition

Version 3 Published

Forms & UI

Processes

Properties & Export

Performance

Logs

Notes Environment Variable Message Re

version 3 2019102
- Added deadline to escalate and send reminder

version 2 20191018
- Added email tool
- Added datalist and userview for end user to access

version 1 20191015
- Created form and process for purchase requisition

AS NOTED HERE

Keeping Track of App Design Across Different Servers

- With the nature of increment of the last app version when an app is imported in, it is imperative to keep track of the “real” app version (app design).
- Make use of **Notes** in app’s properties.

The screenshot shows the RAD Studio interface for an application named "Purchase Requisition". The left sidebar contains a menu with options: "Design App:", "Forms & UI", "Processes", "Properties & Export" (selected), "Performance", and "Logs". The "Design App:" section shows "Version 3" and "Published" buttons. The main area has tabs for "Notes", "Environment Variable", "Message", and "Resource". The "Notes" tab is active, displaying a list of version history entries:

Line	Notes
1	version 3 2019102
2	=====
3	-Added deadline to escalate and send reminder
4	
5	version 2 20191018
6	=====
7	- Added email tool
8	- Added datalist and userview for end user to access
9	
10	version 1 20191015
11	=====
12	- Created form and process for purchase requisition

Application Published state

- With more than 1 version available for the same Application in a RADS server, it is now possible to toggle between versions.

APP CONFIGURATION MANAGEMENT					
Manage App Version		Delegate App Designer Role		Git Configuration	
	VERSION	PUBLISHED	NOTES	DATE CREATED	DATE MODIFIED
	4		version 3 2019102 =====	22-12-2019 09:26 PM	23-12-2019 05:26 AM
	3	✓	-Added deadline to escalate and send reminder version 2 =====	22-12-2019 09:25 PM	23-12-2019 05:34 AM
	2		version 2 20191018 =====	22-12-2019 09:25 PM	23-12-2019 05:33 AM
	1		- Added email tool - Added datalist and userview for end =====	19-12-2019 03:38 PM	23-12-2019 05:33 AM
			version 1 20191015 =====		
			- Created form and process for purchase requisition		

15 Page 1 of 1 Displaying 1 to 4 of 4 items

[New Version](#) [Publish](#) [Unpublish](#) [View](#) [Delete](#)

Published Application Version

- New process instances created will be based on the Published version.
- All elements accessed by end users will also be based on the Published version except for:-
 - For Process Instances created under different Application version, users will continue to use the Forms tied to the specific Application version for its assignments.

Important Note

- By increasing the Application Version, the Process Version will be increased as well.

Exercise on Version Control

1. Create a new RADS Application with a **Process, Form and Userview**. (That's v1)
2. Run the Application, create a new process Instance.
3. Update the Process Design and observe the changes.
4. Increase the Application Version by creating a new version. (From v1 to v2)
5. Modify the Process and Form (in v2), create new process instance and observe the changes.
6. Compare the old and new process instances.

Exercise on Version Control

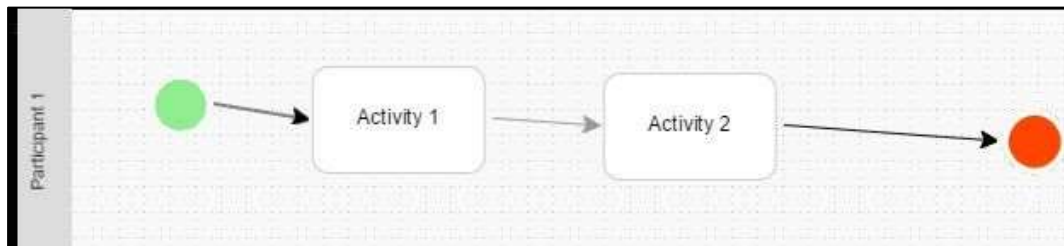
- 1. Create a new RADS Application with a Process, Form and Userview. (That's v1)**
2. Run the Application, create a new process Instance.
3. Update the Process Design and observe the changes.
4. Increase the Application Version by creating a new version. (From v1 to v2)
5. Modify the Process and Form (in v2), create new process instance and observe the changes.
6. Compare the old and new process instances.

Exercise on Version Control

1. Create a new RADS Application with a **Process, Form and Userview**. (That's v1)

Example:

A process flow with 2 activities with both of them mapped to the same form that contains 2 text fields.



Sample	
Title	<input type="text"/>
Description	<input type="text"/>

Drag This Column

Version Control

Click to edit

#date.EEE, d MMM yyyy#

Menu

<i class='icon-home'></i> Home

Drop menu item here

Welcome

Run Process

Inbox

Exercise on Version Control

1. Create a new RADS Application with a **Process, Form and Userview**. (That's v1)
2. **Run the Application, create a new process Instance.**
3. Update the Process Design and observe the changes.
4. Increase the Application Version by creating a new version. (From v1 to v2)
5. Modify the Process and Form (in v2), create new process instance and observe the changes.
6. Compare the old and new process instances.

Exercise on Version Control

2. Run the Application, create a new process Instance.

The screenshot shows the 'VERSION CONTROL' interface. The left sidebar contains a user profile 'Admin Admin' and navigation links for 'Home', 'Run Process', and 'Inbox'. The main content area is titled 'PROCESS 1' and contains a form with the following fields:

- Sample**
- Title**: Test App Version 1
- Description**

Below the form is an orange 'Submit' button. A red arrow points from this button to a table in the bottom right corner. The table has two columns: 'ACTIVITY NAME' and 'PROCESS NAME'.

ACTIVITY NAME	PROCESS NAME
Activity 2	Process 1

Exercise on Version Control

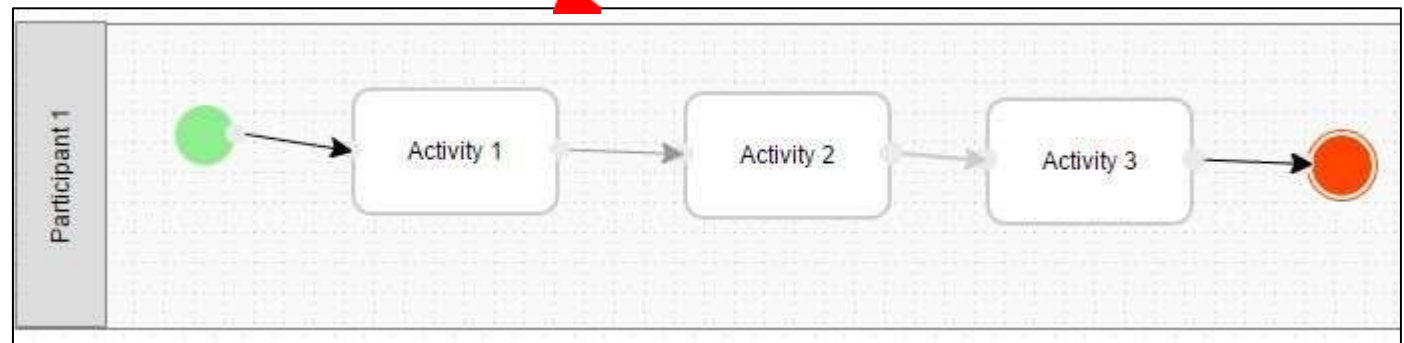
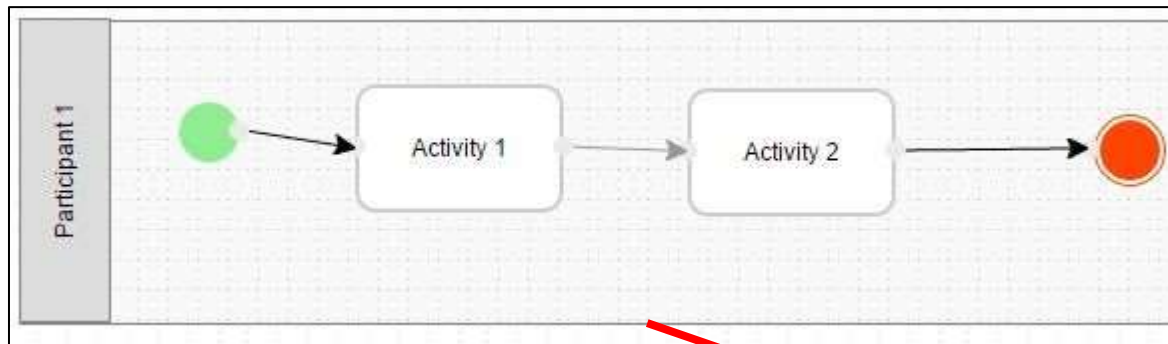
2. Run the Application, create a new process Instance.
 - Observe that on the completion of Activity 1, it will flow to Activity 2.
 - On completion of Activity 2, the process instance comes to an end.
 - Create another process instance and have the it pending at Activity 2 to proceed to the next step.

Exercise on Version Control

1. Create a new RADS Application with a **Process, Form and Userview**. (That's v1)
2. Run the Application, create a new process Instance.
- 3. Update the Process Design and observe the changes.**
4. Increase the Application Version by creating a new version. (From v1 to v2)
5. Modify the Process and Form (in v2), create new process instance and observe the changes.
6. Compare the old and new process instances.

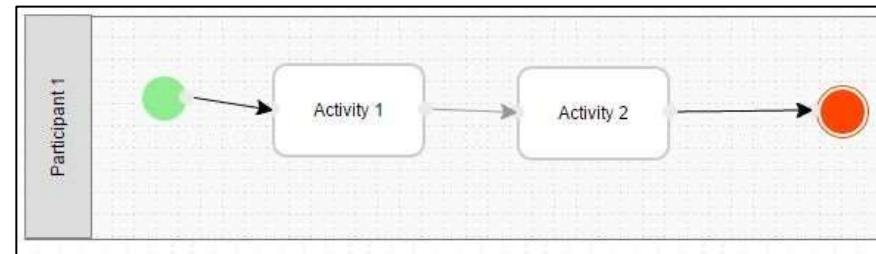
Exercise on Version Control

3. Update the Process Design and observe the changes.

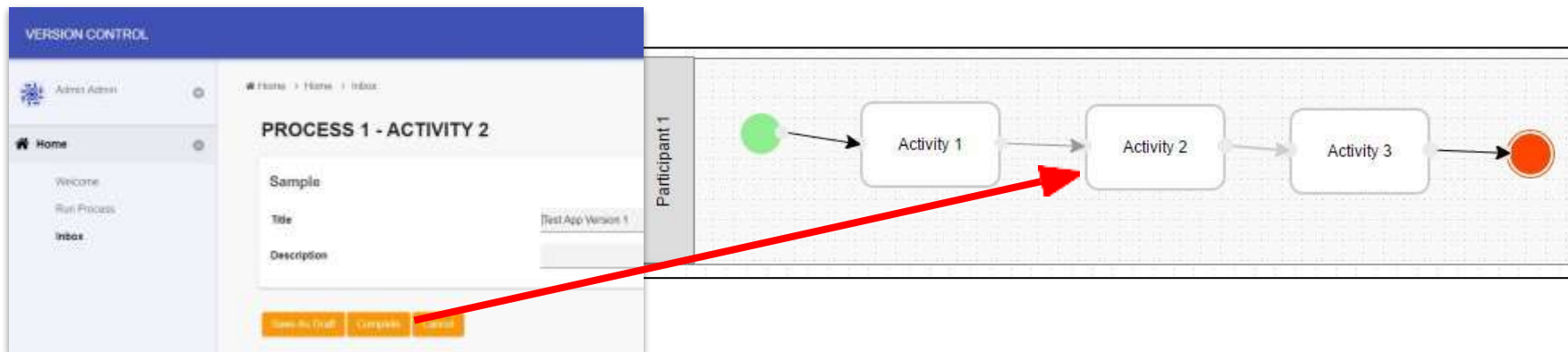


Exercise on Version Control

3. Update the Process Design and observe the changes.
 - Observe that we have process instance that is started before the process design change.



- On completion of Activity 2, what will happen?



Exercise on Version Control

1. Create a new RADS Application with a **Process, Form and Userview**. (That's v1)
2. Run the Application, create a new process Instance.
3. Update the Process Design and observe the changes.
- 4. Increase the Application Version by creating a new version. (From v1 to v2)**
5. Modify the Process and Form (in v2), create new process instance and observe the changes.
6. Compare the old and new process instances.

Exercise on Version Control

4. Increase the Application Version by creating a new version. (From v1 to v2)

The image shows a workflow editor interface. At the top, a process flow is visible with three activities: Activity 1, Activity 2, and Activity 3, connected by arrows. A red box highlights this flow. Below the flow, a 'Version Control' dialog is open, showing a 'Sample' form with fields for 'Title' and 'Description'. A red arrow points from the 'Version Control' dialog to the 'Sample' form. The background shows a menu with items like 'Home', 'Welcome', 'Run Process', and 'Inbox'. The 'Version Control' dialog also shows a 'Menu' section with a 'Home' button and a 'Drop menu item here' label.

Participant 1

Activity 1 → Activity 2 → Activity 3

Version Control
Click to edit
#date.EEE, d MMM yyyy#

Menu

<i class='icon-home'></i> Home

Welcome

Run Process

Inbox

Sample
Drag This Column
Title
Description

Version Control
Click to edit
#date.EEE, d MMM yyyy#

Menu

<i class='icon-home'></i> Home

Welcome

Run Process

Inbox

Apps and Workflow Made Simple
Build apps, not just processes!
Create full-fledged apps with support for data records management.

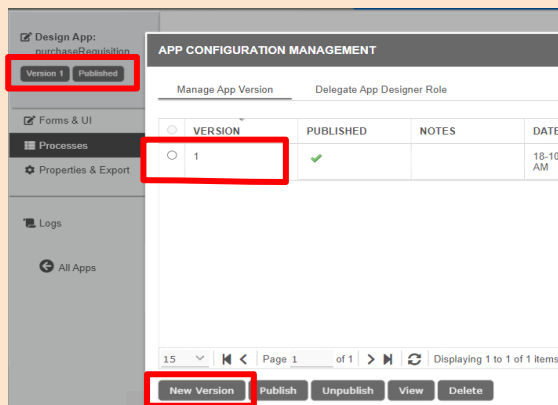
Increased Efficiency and Productivity
Faster and more consistent completion of manual processes, and with minimal errors.

Lowered Cost
Employees can be guided through complex procedures, hence reducing the cost of training.

Exercise on Version Control

4. Increase the Application Version by creating a new version. (From v1 to v2)
 - Observe that at this point of time, App Version 1 and App Version 2 are identical.
 - Switch the published version from 1 to 2.

RECAP: App Designer > Versions > Select version > New Version



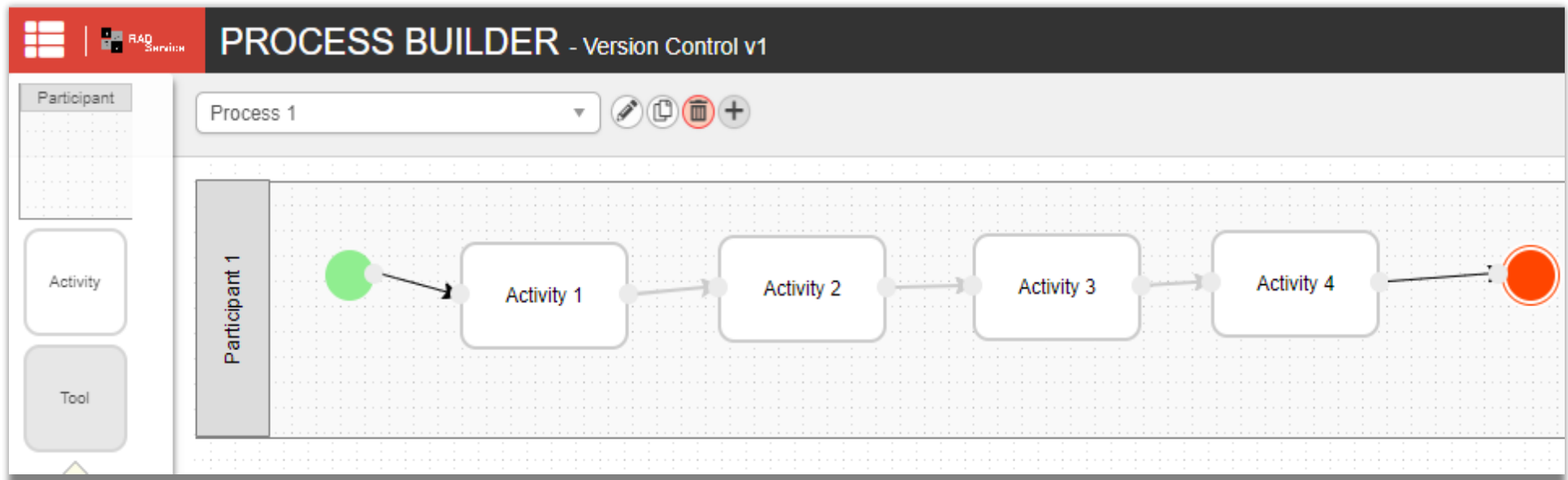
By creating a **New Version**, the App design will be **cloned** into the new version.

Exercise on Version Control

1. Create a new RADS Application with a **Process, Form and Userview**. (That's v1)
2. Run the Application, create a new process Instance.
3. Update the Process Design and observe the changes.
4. Increase the Application Version by creating a new version. (From v1 to v2)
5. **Modify the Process and Form (in v2), create new process instance and observe the changes.**
6. Compare the old and new process instances.

Exercise on Version Control

5. Modify the Process and Form (in v2), create new process instance and observe the changes.
 - Add new text field to the form.
 - Add new activity to the process.



Exercise on Version Control

1. Create a new RADS Application with a **Process, Form and Userview**. (That's v1)
2. Run the Application, create a new process Instance.
3. Update the Process Design and observe the changes.
4. Increase the Application Version by creating a new version. (From v1 to v2)
5. Modify the Process and Form (in v2), create new process instance and observe the changes.
6. **Compare the old and new process instances.**

Exercise on Version Control

6. Compare the old and new process instances.
 - Will process instances started on **App Version 2** flow to **Activity 4**?
 - Will process instances started on **App Version 1** flow to **Activity 4**?
 - Which process instance is showing the new form design, why and why not?

Exercise on Version Control

1. Create a new RADS Application with a **Process, Form and Userview**. (That's v1)
2. Run the Application, create a new process Instance.
3. Update the Process Design and observe the changes.
4. Increase the Application Version by creating a new version. (From v1 to v2)
5. Modify the Process and Form (in v2), create new process instance and observe the changes.
6. Compare the old and new process instances.

Lessons Learnt From The Exercise

- Changing process design of App Version 2 did NOT affect running instance of App Version 1.
- Each App Version would only contain the one (and latest) process design.
- Running instances of App Version 1 will show Forms of App Version 1, likewise, for Version 2, regardless of current Published App Version.
- Forms will be shown based on Published App Version except for those tied to running instances.

Chapter Review

- Understand how to manage Application Version and its impact.

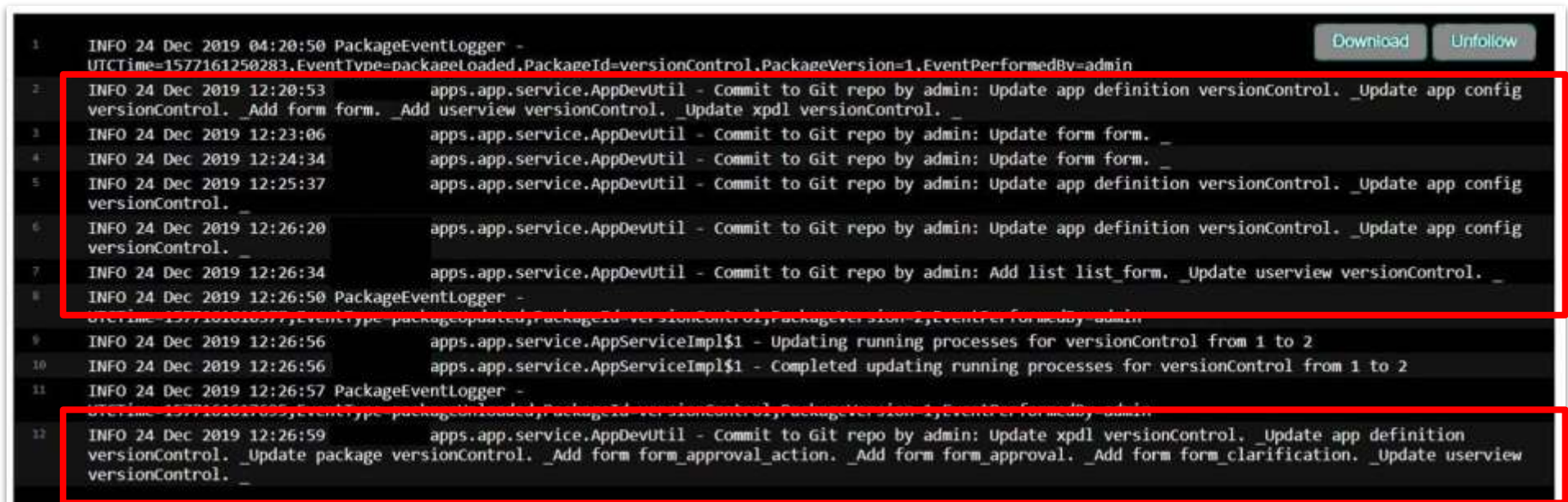


Chapter 4

Git Version Control

Built-in Git

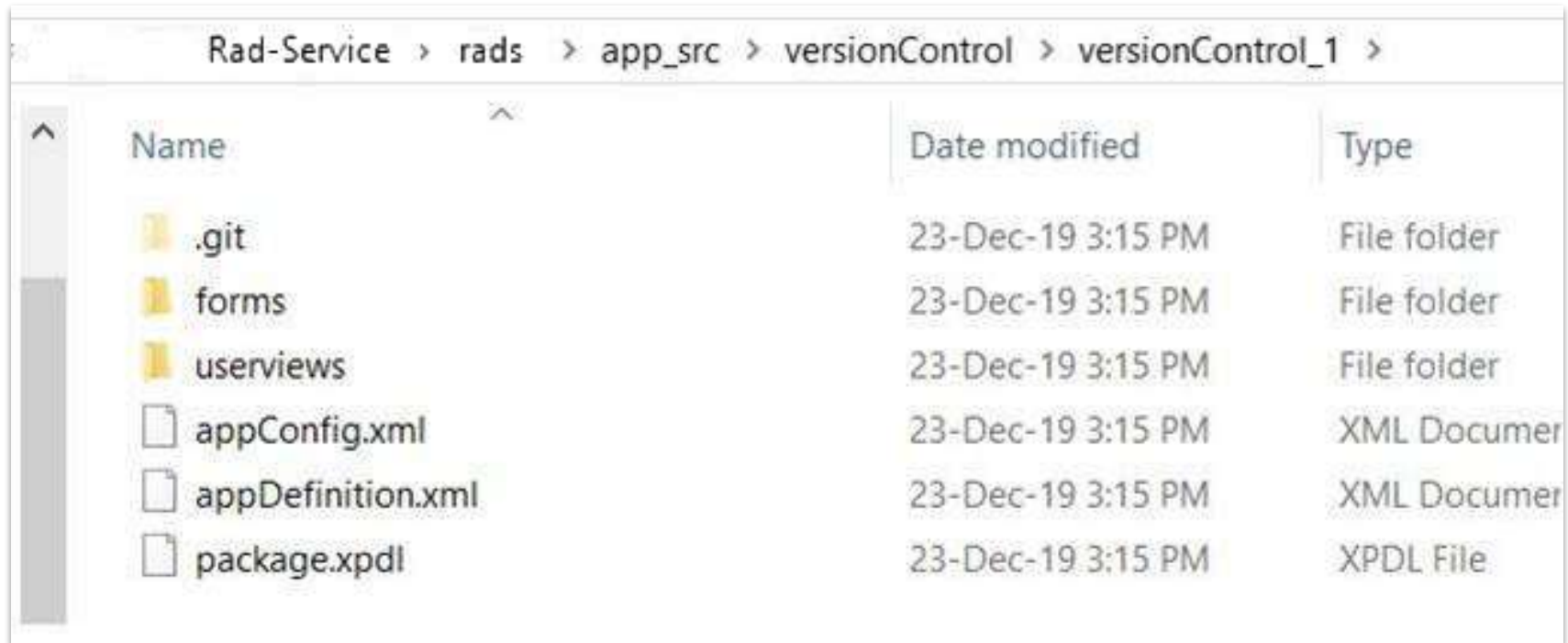
- Any changes within the app will be committed into Git



```
1 INFO 24 Dec 2019 04:20:50 PackageEventLogger -  
   UTCTime=1577161250283,EventType=packageLoaded,PackageId=versionControl,PackageVersion=1,EventPerformedBy=admin  
2 INFO 24 Dec 2019 12:20:53 apps.app.service.AppDevUtil - Commit to Git repo by admin: Update app definition versionControl. _Update app config  
   versionControl. _Add form form. _Add userview versionControl. _Update xpd versionControl. _  
3 INFO 24 Dec 2019 12:23:06 apps.app.service.AppDevUtil - Commit to Git repo by admin: Update form form. _  
4 INFO 24 Dec 2019 12:24:34 apps.app.service.AppDevUtil - Commit to Git repo by admin: Update form form. _  
5 INFO 24 Dec 2019 12:25:37 apps.app.service.AppDevUtil - Commit to Git repo by admin: Update app definition versionControl. _Update app config  
   versionControl. _  
6 INFO 24 Dec 2019 12:26:20 apps.app.service.AppDevUtil - Commit to Git repo by admin: Update app definition versionControl. _Update app config  
   versionControl. _  
7 INFO 24 Dec 2019 12:26:34 apps.app.service.AppDevUtil - Commit to Git repo by admin: Add list list_form. _Update userview versionControl. _  
8 INFO 24 Dec 2019 12:26:50 PackageEventLogger -  
   UTCTime=1577161610377,EventType=packageUpdated,PackageId=versionControl,PackageVersion=2,EventPerformedBy=admin  
9 INFO 24 Dec 2019 12:26:56 apps.app.service.AppServiceImpl$1 - Updating running processes for versionControl from 1 to 2  
10 INFO 24 Dec 2019 12:26:56 apps.app.service.AppServiceImpl$1 - Completed updating running processes for versionControl from 1 to 2  
11 INFO 24 Dec 2019 12:26:57 PackageEventLogger -  
   UTCTime=1577161617059,EventType=packageUnloaded,PackageId=versionControl,PackageVersion=1,EventPerformedBy=admin  
12 INFO 24 Dec 2019 12:26:59 apps.app.service.AppDevUtil - Commit to Git repo by admin: Update xpd versionControl. _Update app definition  
   versionControl. _Update package versionControl. _Add form form_approval_action. _Add form form_approval. _Add form form_clarification. _Update userview  
   versionControl. _
```

How To Access the Built-In Git?

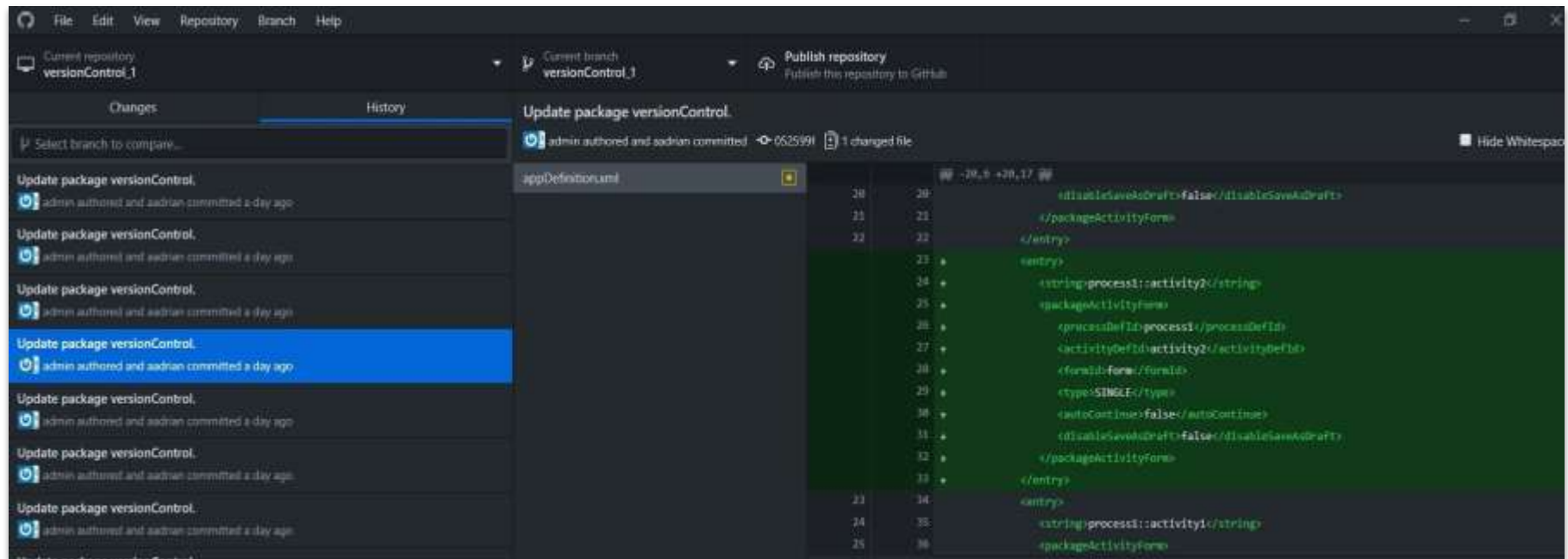
- To access into the built-in Git, the local repository is in <RADS installation folder>\rads\app_src\<App ID>\<App ID_version number>



Rad-Service > rads > app_src > versionControl > versionControl_1 >			
^	Name	Date modified	Type
	.git	23-Dec-19 3:15 PM	File folder
	forms	23-Dec-19 3:15 PM	File folder
	userviews	23-Dec-19 3:15 PM	File folder
	appConfig.xml	23-Dec-19 3:15 PM	XML Document
	appDefinition.xml	23-Dec-19 3:15 PM	XML Document
	package.xpdl	23-Dec-19 3:15 PM	XPDL File

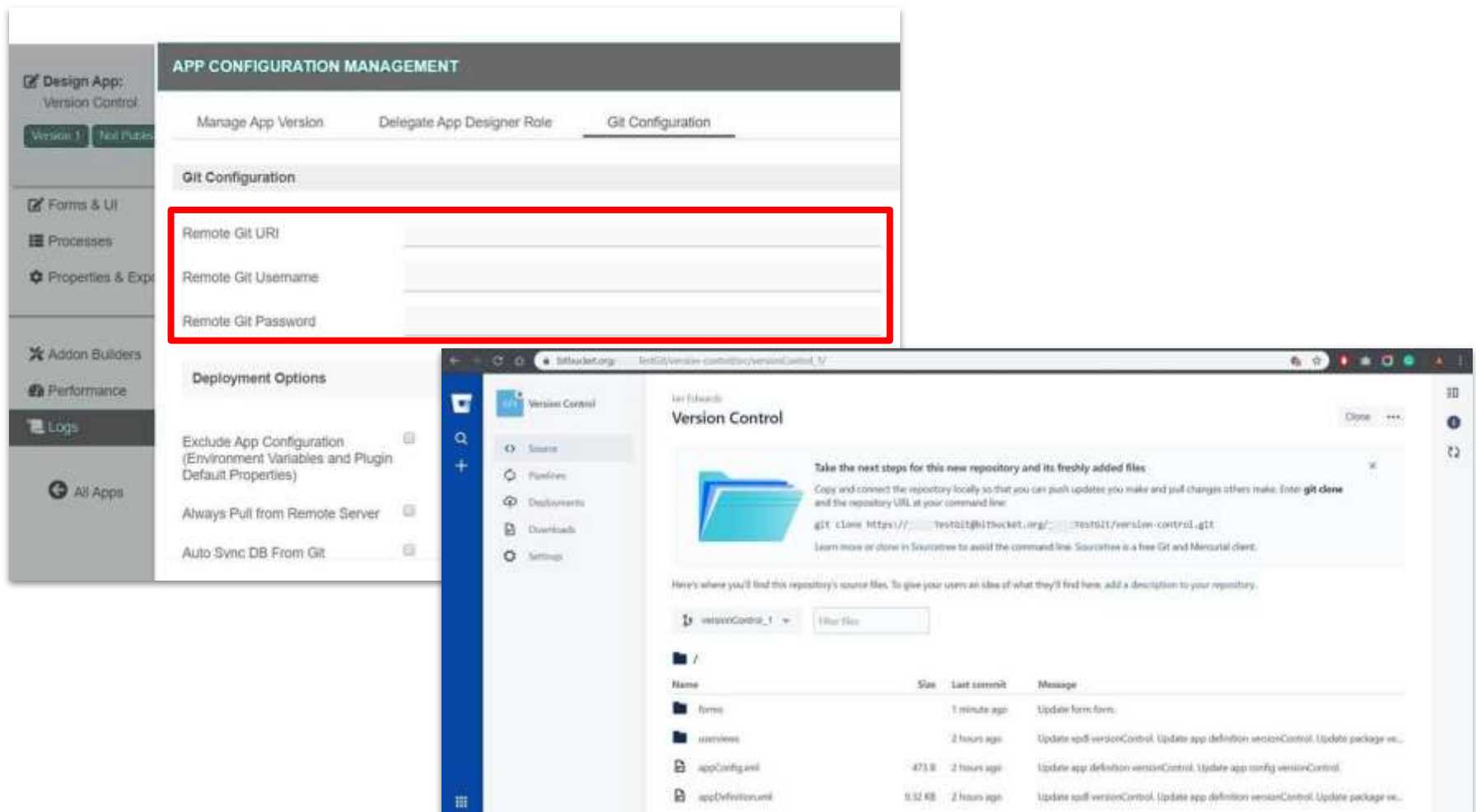
How To Access the Built-In Git?

- Sample of using GitHub Desktop to access the Git Repository



Remote Git

- You can also integrate to external Git



Module Review

1. Introduction to Version Control
2. Process Version Control
3. Application Version Control
4. Git Version Control

Recommended Further Learning

- <https://docs.rads.purwana.net/Version>

Stay Connected With RADS

- rads.purwana.net
- <https://github.com/radservice/rad-community>