

# Snapshot Best Practices: Change and Release Management

## Introduction

Managing a large and complex Salesforce org can be a daunting task. Change and release challenges include moving metadata between multiple environments, unexpected Sandbox refreshes, different Salesforce account types, Salesforce API version skew, dealing with manual changes, and handling large numbers of metadata assets. Native tools such as Change Sets, ANT, Force.com IDE, and the Setup Menu require lots of time and resources.

Snapshot was designed to make the complex task of managing a large Salesforce org much easier. The product was first released on the AppExchange in 2006. The engineering team at DreamFactory and later Metazoa has updated the product through 30 versions of the Metadata API. Many of Salesforce's largest customers use Snapshot for change and release management. In 2018 Metazoa released a major upgrade to the product which embodies more than a decade of accumulated wisdom about how to simplify the task of change and release management for Salesforce administrators.

Snapshot provides highly visual tools for retrieving metadata from any Salesforce account or developer repository. Metadata assets can be moved into a partial or full Sandbox, and after quality assurance and user acceptance testing, deployed into a production org. As part of the migration process, Snapshot provides two dozen security and compliance reports that help you analyze metadata structure, data usage, and account security. This whitepaper is a guide to the best practices for using Snapshot to conduct change and release management operations across a series of connected Salesforce accounts.

## Institutional Foundation

Before we get started, let's have a brief discussion about the institutions necessary for an effective change and release governance framework. Here are some of the best practices that you may be considering or have already implemented at your company:

- Establish a Center of Excellence to ensure that changes support business goals
- Create a release management process for daily, minor, and major changes
- Sign-off on every release phase and leave an audit trail for compliance
- Automate org monitoring for metadata changes, security problems, and errors
- Create a backlog and priority list that is coordinated with a release calendar
- Implement a pre and post deployment checklists for manual changes
- Use design standards for naming, descriptions, deprecation, and testing

Salesforce provides a useful guide on managing change with a governance framework:

[https://trailhead.salesforce.com/modules/app\\_deployment/units/app\\_deployment\\_governance](https://trailhead.salesforce.com/modules/app_deployment/units/app_deployment_governance)

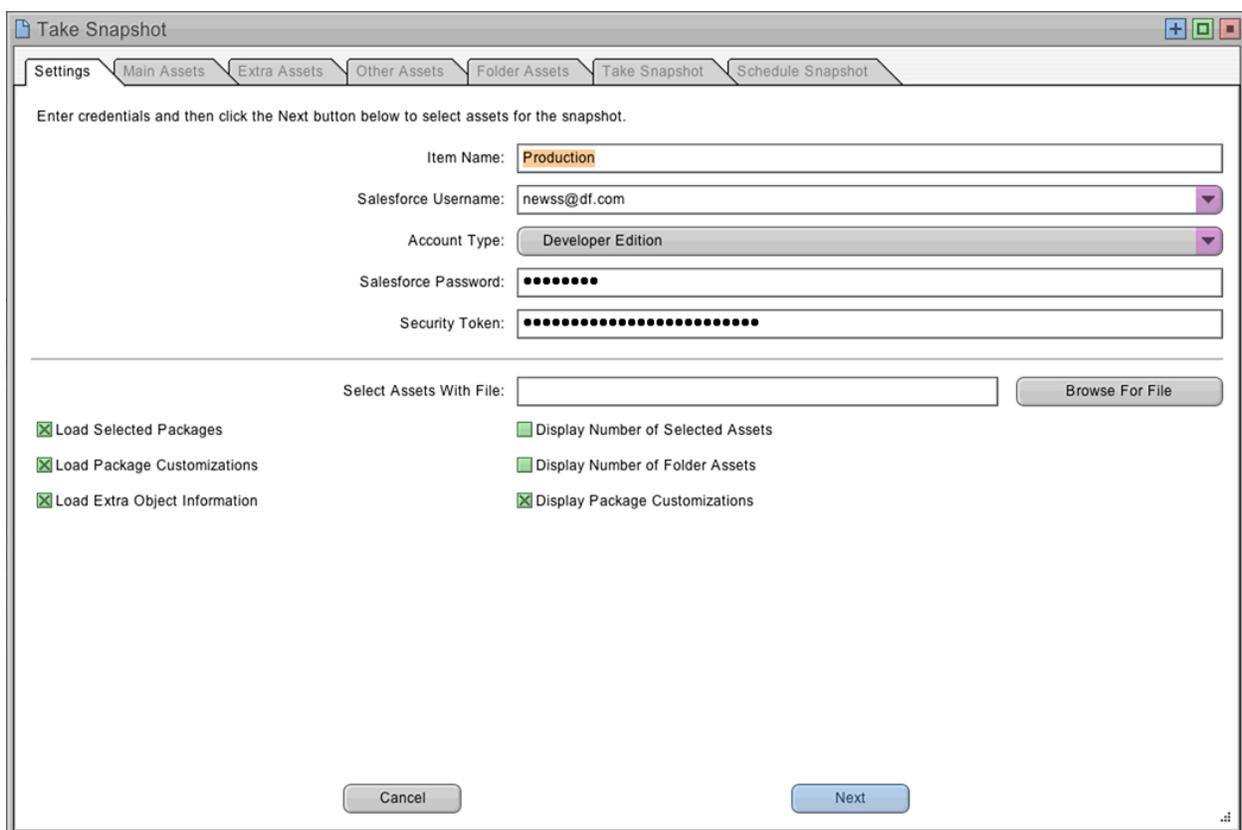
Many of the capabilities in Snapshot are designed to support corporate governance, compliance, and security practices. Here are some of the features that you can take advantage of for this purpose:

- Metadata time series backup addresses compliance
- Deployment History provides governance for migrations
- Data Dictionary report documents Object and Field properties
- Metadata Differences and Code Coverage reports monitor org changes
- Org Limits and Security Health Check reports trigger security alerts
- Setup Audit Trail used to validate metadata prior to deployment
- Team coordination and automation with Chatter, Content and Email
- Job Lists used for migration are saved and shared across team
- Activity Timeline tracks user actions over defined time frame

## Taking Snapshots

The Take Snapshot dialog allows you to select the metadata assets that you want to work with. These assets can be used for reporting or migration to another org. Assets that are very numerous or very large are selected individually. These assets include: Packages, Profiles, Permission Sets, Roles, Groups, Queues, Custom Object Translations, Page Layouts, Apex Classes, Apex Pages, Dashboards, Reports, Email Templates, and Documents. All other metadata assets types are available in a “catch all” list for quick selection.

The assets that you select for the source snapshot will be available for migration to the destination. The assets that you select for the destination snapshot will be available for deletion. The destination assets are also important because they show the visual differences between the source and destination.



Take Snapshot

Settings Main Assets Extra Assets Other Assets Folder Assets Take Snapshot Schedule Snapshot

Enter credentials and then click the Next button below to select assets for the snapshot.

Item Name:

Salesforce Username:

Account Type:

Salesforce Password:

Security Token:

Select Assets With File:

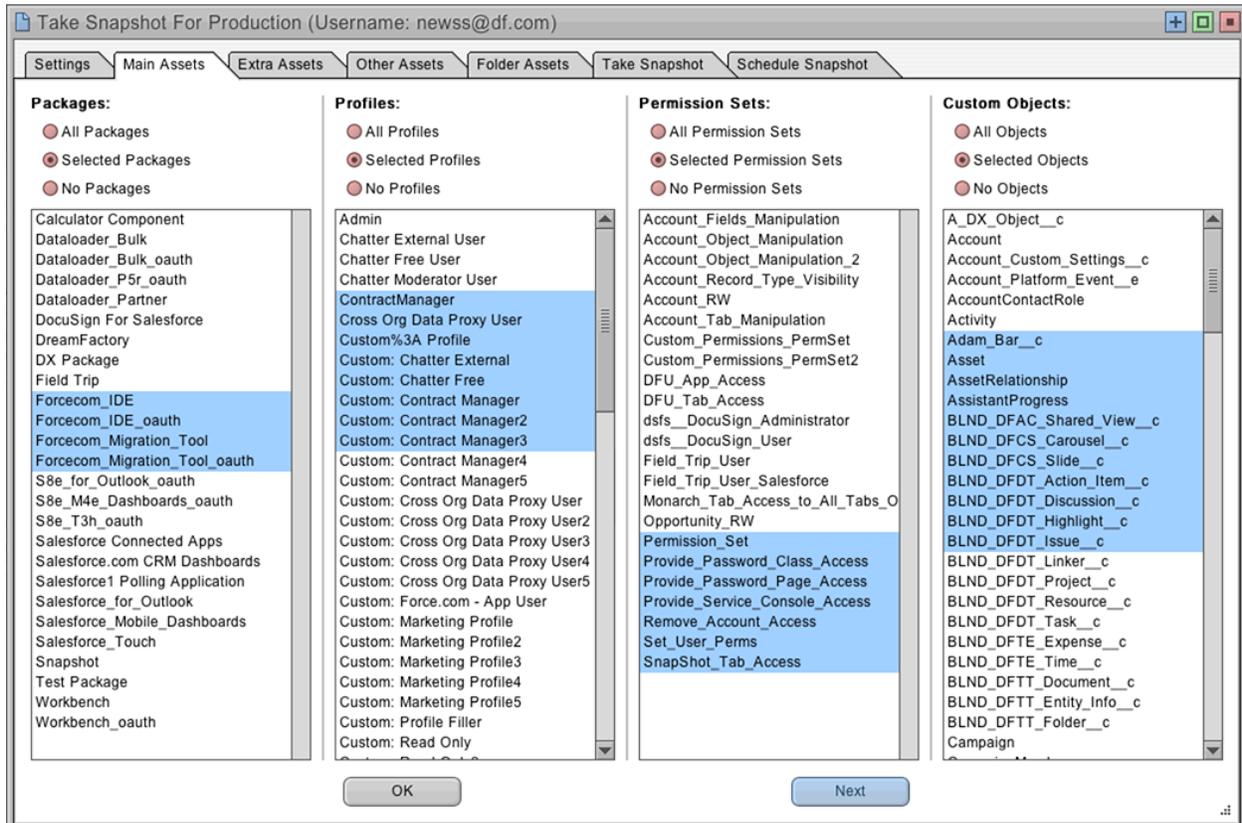
Load Selected Packages  Display Number of Selected Assets

Load Package Customizations  Display Number of Folder Assets

Load Extra Object Information  Display Package Customizations

There is a limit of 10,000 assets that can be downloaded in a single Metadata API transaction. For smaller orgs and Developer Sandboxes, you may be able to simply download all the Metadata. For larger orgs, you will need a strategy to decide which assets to download and work with. Try to select a superset of assets that cover your deployment needs without downloading more information than necessary. Perhaps you can discriminate on the basis of developer projects or Packages. This saves time and simplifies deployments and reporting.

Profiles will be a problem for some deployments. The size of each Profile depends on what other Apex Classes, Apex Pages, Custom Applications, Custom Objects, Page Layouts, and Custom Tabs are included in the snapshot. In particular, Field Permissions can take up huge amounts of space. Imagine that you have 500 Profiles and 500 Objects each with 500 fields. In this example, the Metadata API would retrieve over a million Field Permission junctions. While this may be possible, gigantic Profiles are difficult to work with and take lots of time to retrieve and deploy. Try to cut down either the number of Profiles or Objects needed for the task at hand.

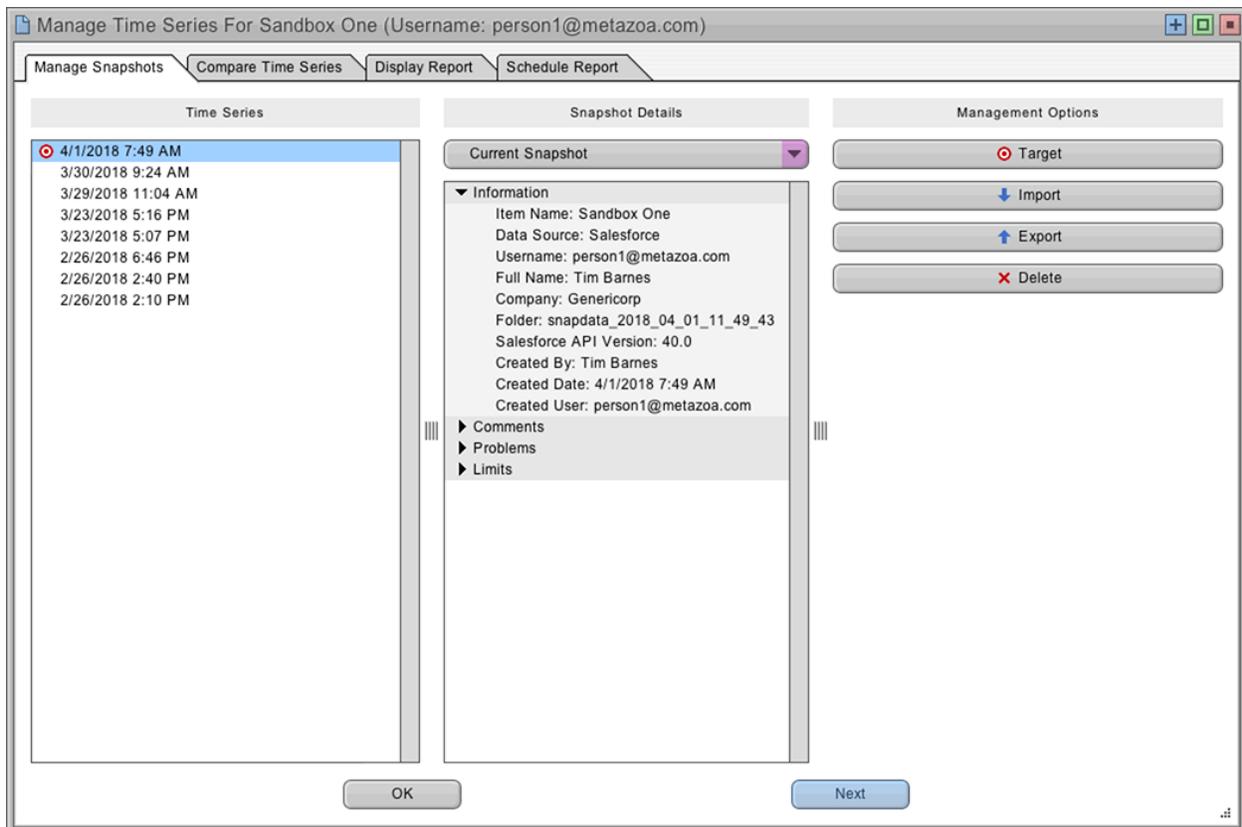


The “Select Assets With File” interface on the Take Snapshot dialog can help load only the metadata Assets needed for a particular job. You can browse for any Job List or Package.xml file and automatically select those assets. Often a developer working on a project will send a file along to call out the assets that need to be migrated. The same Job List can be imported into the “Deploy Metadata” dialog to move the same assets later on.

## Time Series Data

Each time you take a new snapshot, the old snapshot is not destroyed. In fact, all of the previous snapshots are maintained in a time series. You can use the Manage Time Series dialog to view all of the recorded snapshots through time. Any one of the older snapshots can be targeted for reporting or migration. Simply select the desired snapshot and click the Target button. The automatic storage of time series snapshots is a key feature for compliance.

All of the differences across the time series can be viewed and reported on in the Compare Time Series tab. This provides an easy way to actively monitor any section of metadata for changes. If a change is detected, a comprehensive difference report can be mailed out to team members. This capability can be used to document progress on a specific development project or trigger an alert when metadata is changed for any reason.



Manage Time Series For Sandbox One (Username: person1@metazoa.com)

Manage Snapshots | Compare Time Series | Display Report | Schedule Report

Time Series

- 4/1/2018 7:49 AM
- 3/30/2018 9:24 AM
- 3/29/2018 11:04 AM
- 3/23/2018 5:16 PM
- 3/23/2018 5:07 PM
- 2/26/2018 6:46 PM
- 2/26/2018 2:40 PM
- 2/26/2018 2:10 PM

Snapshot Details

Current Snapshot

Information

- Item Name: Sandbox One
- Data Source: Salesforce
- Username: person1@metazoa.com
- Full Name: Tim Barnes
- Company: Genericorp
- Folder: snapdata\_2018\_04\_01\_11\_49\_43
- Salesforce API Version: 40.0
- Created By: Tim Barnes
- Created Date: 4/1/2018 7:49 AM
- Created User: person1@metazoa.com

- Comments
- Problems
- Limits

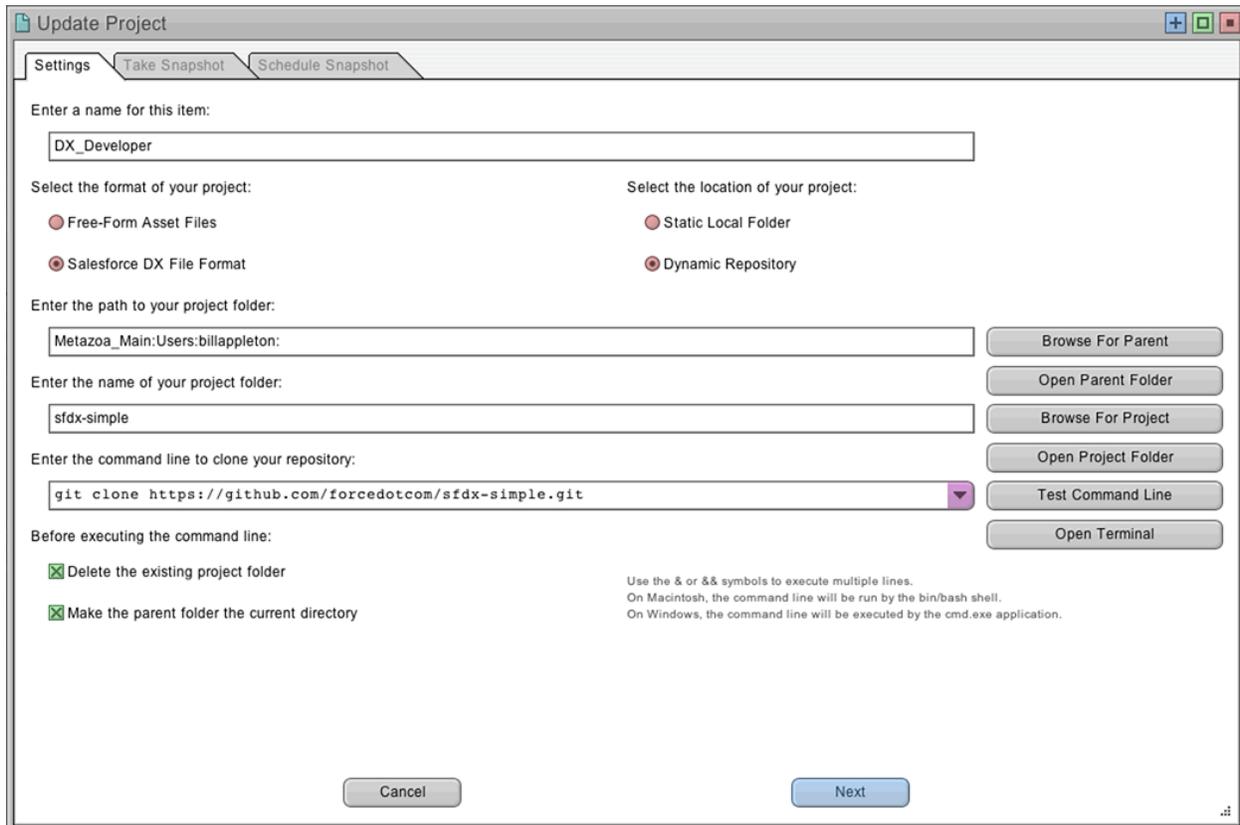
Management Options

- Target
- Import
- Export
- Delete

OK Next

## Developer Projects

The Take Snapshot dialog can download metadata assets from any Salesforce org. But sometimes metadata assets reside in an online repository or a local file folder. A change and release manager might receive developer assets by email and save them into a folder. Or a developer could provide a link to an online repository with their latest project. The folder can contain a Salesforce DX project or a free-form collection of asset files. The Update Project dialog can handle all of these situations.



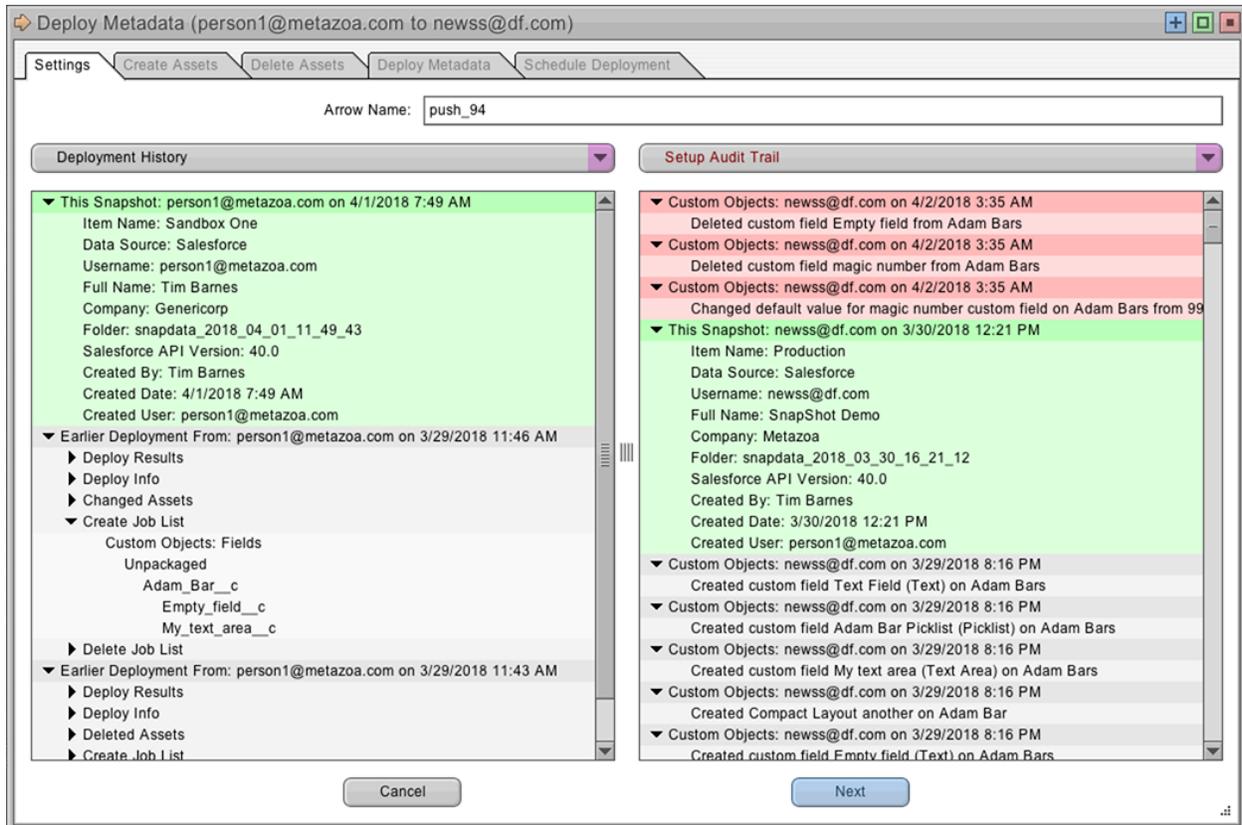
The free-form asset folder is like a shoebox full of loose metadata asset files and other folders. The layout can be the same as the Force.com IDE, the Metadata API, or some other format. Snapshot will parse the file and folder structure and figure out how to harvest the metadata. A Salesforce DX project will also have a well-defined format that can be converted to a snapshot.

The project folder can be located on the local machine or an online repository. You will need a command line interface to download the repository. Snapshot provides example command lines for Git and Subversion. The release manager can use the same command line interface that their developer uses to clone the project. The command line can be tested from the Update Project dialog.

When you update a developer project the files are downloaded from the repository (if any) and then harvested just like a regular snapshot. The developer project can be connected and pushed into a Salesforce org as well. One difference is that, because the developer project is not connected to an actual Salesforce account, some data usage and security reports are not available.

## Deploying Changes

Once you have a source and destination snapshot, you can double-click the deployment arrow to move metadata from the source to the destination org. The first screen you see will show you the state of the source and destination snapshot. This includes Setup Audit Trail and Deployment History information. If the source or destination snapshot is out of date, you can take another snapshot immediately from this interface. This capability guarantees that multiple team members working in an active org are not stepping on each other's metadata, and also that the current snapshot being deployed and the destination snapshot being compared are up to date.



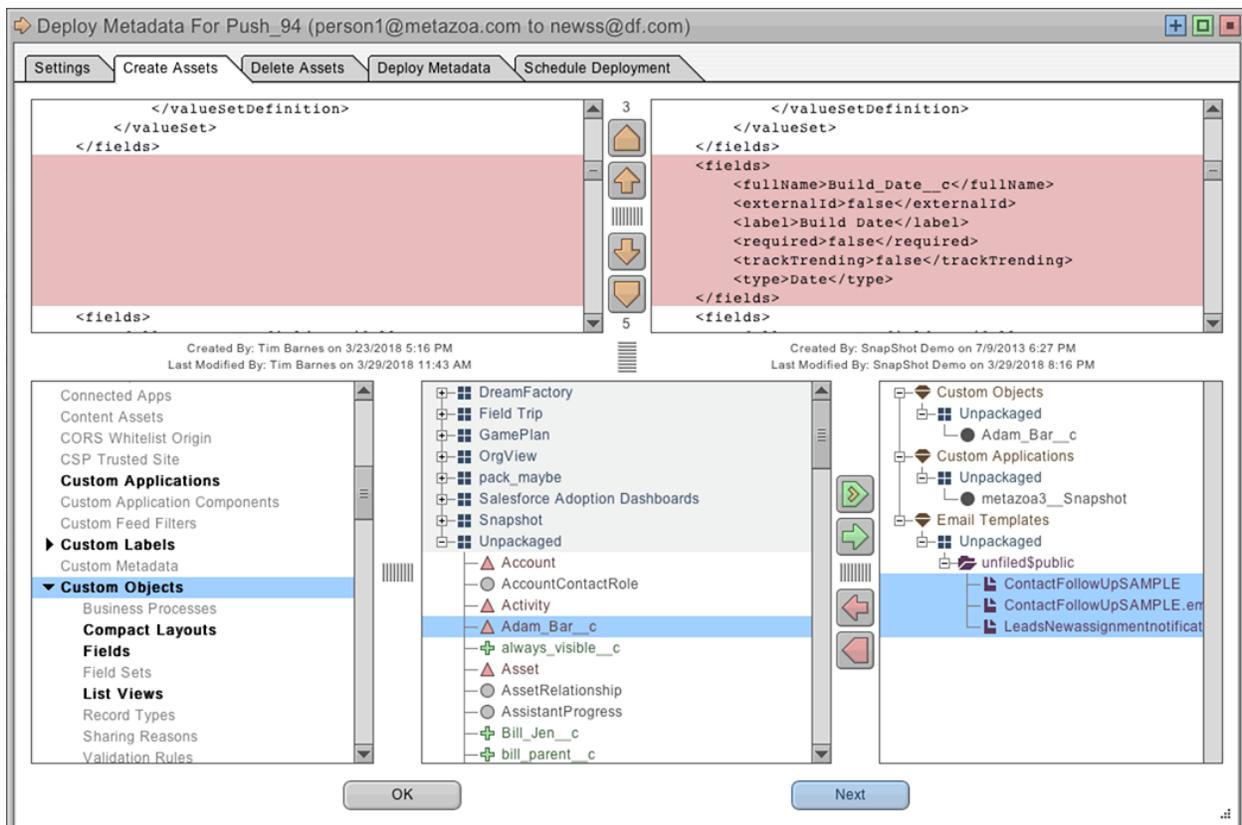
The screenshot shows the 'Deploy Metadata' window with the following details:

- Window Title:** Deploy Metadata (person1@metazoa.com to newss@df.com)
- Navigation:** Settings, Create Assets, Delete Assets, Deploy Metadata, Schedule Deployment
- Arrow Name:** push\_94
- Deployment History:**
  - This Snapshot: person1@metazoa.com on 4/1/2018 7:49 AM**
    - Item Name: Sandbox One
    - Data Source: Salesforce
    - Username: person1@metazoa.com
    - Full Name: Tim Barnes
    - Company: Genericorp
    - Folder: snapdata\_2018\_04\_01\_11\_49\_43
    - Salesforce API Version: 40.0
    - Created By: Tim Barnes
    - Created Date: 4/1/2018 7:49 AM
    - Created User: person1@metazoa.com
  - Earlier Deployment From: person1@metazoa.com on 3/29/2018 11:46 AM**
    - Deploy Results
    - Deploy Info
    - Changed Assets
    - Create Job List
      - Custom Objects: Fields
        - Unpackaged
          - Adam\_Bar\_\_c
          - Empty\_field\_\_c
          - My\_text\_area\_\_c
      - Delete Job List
    - Earlier Deployment From: person1@metazoa.com on 3/29/2018 11:43 AM**
      - Deploy Results
      - Deploy Info
      - Deleted Assets
      - Create Job List
- Setup Audit Trail:**
  - Custom Objects: newss@df.com on 4/2/2018 3:35 AM**
    - Deleted custom field Empty field from Adam Bars
  - Custom Objects: newss@df.com on 4/2/2018 3:35 AM**
    - Deleted custom field magic number from Adam Bars
  - Custom Objects: newss@df.com on 4/2/2018 3:35 AM**
    - Changed default value for magic number custom field on Adam Bars from 99
  - This Snapshot: newss@df.com on 3/30/2018 12:21 PM**
    - Item Name: Production
    - Data Source: Salesforce
    - Username: newss@df.com
    - Full Name: SnapShot Demo
    - Company: Metazoa
    - Folder: snapdata\_2018\_03\_30\_16\_21\_12
    - Salesforce API Version: 40.0
    - Created By: Tim Barnes
    - Created Date: 3/30/2018 12:21 PM
    - Created User: person1@metazoa.com
  - Custom Objects: newss@df.com on 3/29/2018 8:16 PM**
    - Created custom field Text Field (Text) on Adam Bars
  - Custom Objects: newss@df.com on 3/29/2018 8:16 PM**
    - Created custom field Adam Bar Picklist (Picklist) on Adam Bars
  - Custom Objects: newss@df.com on 3/29/2018 8:16 PM**
    - Created custom field My text area (Text Area) on Adam Bars
  - Custom Objects: newss@df.com on 3/29/2018 8:16 PM**
    - Created Compact Layout another on Adam Bar
  - Custom Objects: newss@df.com on 3/29/2018 8:16 PM**
    - Created custom field Empty field (Text) on Adam Bars

Next, you will need to build Job Lists for the metadata assets that you want to create and delete. The Metadata API always deletes and then creates some assets in each transaction, so that is why the user interface is organized in this fashion. You can also just create or just delete assets if desired.

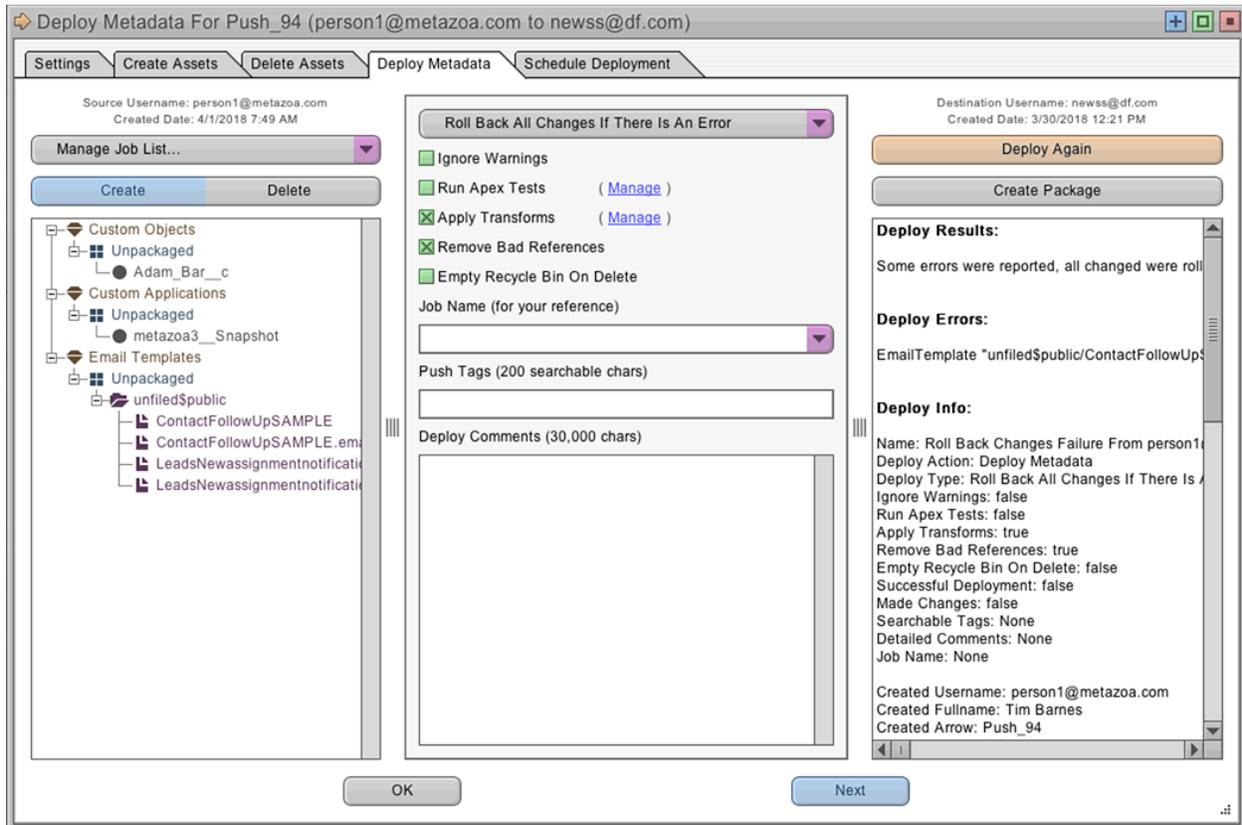
The left-hand column shows all of the metadata types that are available. The disabled types were not captured by the current snapshot. The middle column shows the assets for the selected type that are available to be added to the Job List. The right-hand column shows the assets that have already been selected for the Job List.

Notice that the middle column marks each asset with a grey circle, red triangle, or green plus icon. The assets marked with a grey circle are the same in the source and destination. The assets marked with a red triangle are different in the source and destination. The assets marked with a green plus are only available on the source and are missing from the destination. The slide-down panel at the top of the screen shows the line by line differences between the currently selected assets.



The buttons in the lower right change based on what is selected. You can move an asset into the Job List, remove an asset, move all assets, or remove all assets. Other options include the ability to move only different or missing assets. Lastly, you can move an assets and all related metadata types. This is useful for grabbing additional types that would otherwise cause the deployment to fail.

The next tab lets you select assets for deletion. The Create Job List will contain assets gathered from the source snapshot, and the Delete Job List will have assets gathered from the destination. Either one of these Job Lists can be empty. After assembling the Job Lists, the Deploy Metadata tab can be used to push them into the destination org. After each deployment, you can go back to the previous tabs and further refine the Job Lists and continue with additional deployments.



On the left-hand column you can see the Create and Delete Job Lists. The Manage Job Lists button will allow you to import and export Job Lists, as well as browse for previous Job Lists that you or your team members have used while working on this org. Sharing the Create and Delete Job Lists reduces errors and provides a simple way to orchestrate deployments.

The middle column has powerful tools to help with deployment. The Run Apex Tests button allows you to select Apex tests that are run prior to deployment. This is useful for achieving 75% code coverage or setting up a Quick Deploy, discussed below. The Apply Transforms button allows you to transform data values before deployment. For example, you could remap a username from the source to the destination org to enable the successful migration of a Report or Workflow.

The Job Names, Push Tags, and Deploy Comments can be any text string that you want to include for compliance purposes. These values and all of the other information about the migration are written out to the Deployment History custom objects after the push. These objects can also be used for creating reports in the Salesforce HTML interface.

The Remove Bad References checkbox can prevent many problems from happening in a deployment. This option scans Profiles, Permission Sets, Page Layouts, and Custom Objects, and removes any references to assets that are not available in the destination org or included in the current Job List.

Profiles and Permission Sets are scanned for Apex Classes, Apex Pages, Applications, Tabs, Objects, Fields, Layouts, and Record Types that are not available in the destination org or included in the current Job List. Page Layouts are scanned for Fields that are not available in Layout Sections and Related Lists. Custom Objects are scanned for Fields that are not available in Record Types and Compact Layouts.

In each case, all the bad references are seamlessly removed before the metadata asset is deployed. Otherwise, the migration will surely fail. In some cases, like a production deployment, you may want a migration with any error to fail. In other cases, like updating a Sandbox, the option to Remove Bad References can save time and simplify the deployment.

Lastly, the right-hand column has the Deploy Metadata button. Click this button to conduct a Validating, Transactional, or Partial Deployment. When the deployment finishes, there is a human-readable report displayed in the results box below. The results box will also contain information about any references that were removed.

## Deployment Considerations

The menu at the top of the Deploy Metadata dialog provides three options:

**Validating Deployment** – This option will check for errors without making changes. The deployment is simulated, and no changes are actually made to your Salesforce org. All success and error messages are displayed in the results box.

**Transactional Deployment** -- This option will roll back all changes if there is an error. If there are any errors, the entire deployment is rolled back, otherwise all assets will be migrated. All success and error messages are displayed in the results box.

**Partial Deployment** -- This option will change everything possible and ignore errors. Assets that cause errors are rolled back, but other assets continue to deploy unless there are cascading errors. This option is not available for production orgs.

When you are deploying assets to a production org, you must use a Transactional Deployment. This option has the advantage of rolling back all changes if there is an error. Sometimes for more complex migrations, you will need to make multiple deployments. The order of operations for multiple deployments is discussed in the next section.

Here are a few best practices to keep in mind when deploying to a production org:

- Announce the maintenance window for the deployment and expected down time
- Schedule your release to avoid any Salesforce upgrades in the same time period
- Avoid modifying the org with the Setup Menu before a production deployment
- Test with a Validating Deployment to be sure the migration will be error free
- Edit the login hours on Profiles to keep users out during the maintenance window
- Schedule a metadata push or Quick Deploy with Snapshot's automation capabilities

## Order of Operations

We recommend that you push metadata assets in the following order when conducting multiple deployments. This order will minimize the errors generated by contingent dependencies and relationships in your org. Your mileage may vary, but this is a good starting point for a complex deployment. Carefully read any error messages and make adjustments as necessary.

1. Objects \*
2. Apex Classes
3. Apex Components
4. Apex Pages
5. Apex Triggers
6. S-Controls
7. Page Layouts
8. Static Resources
9. Letterheads
10. Workflows \*\*
11. Report Types
12. Home Page Web Links
13. Home Page Components
14. Home Page Layouts
15. Custom Tabs
16. Custom Labels
17. Custom Applications
18. Custom Object Translations
19. Custom Sites
20. Profiles \*\*\*
21. Permission Sets

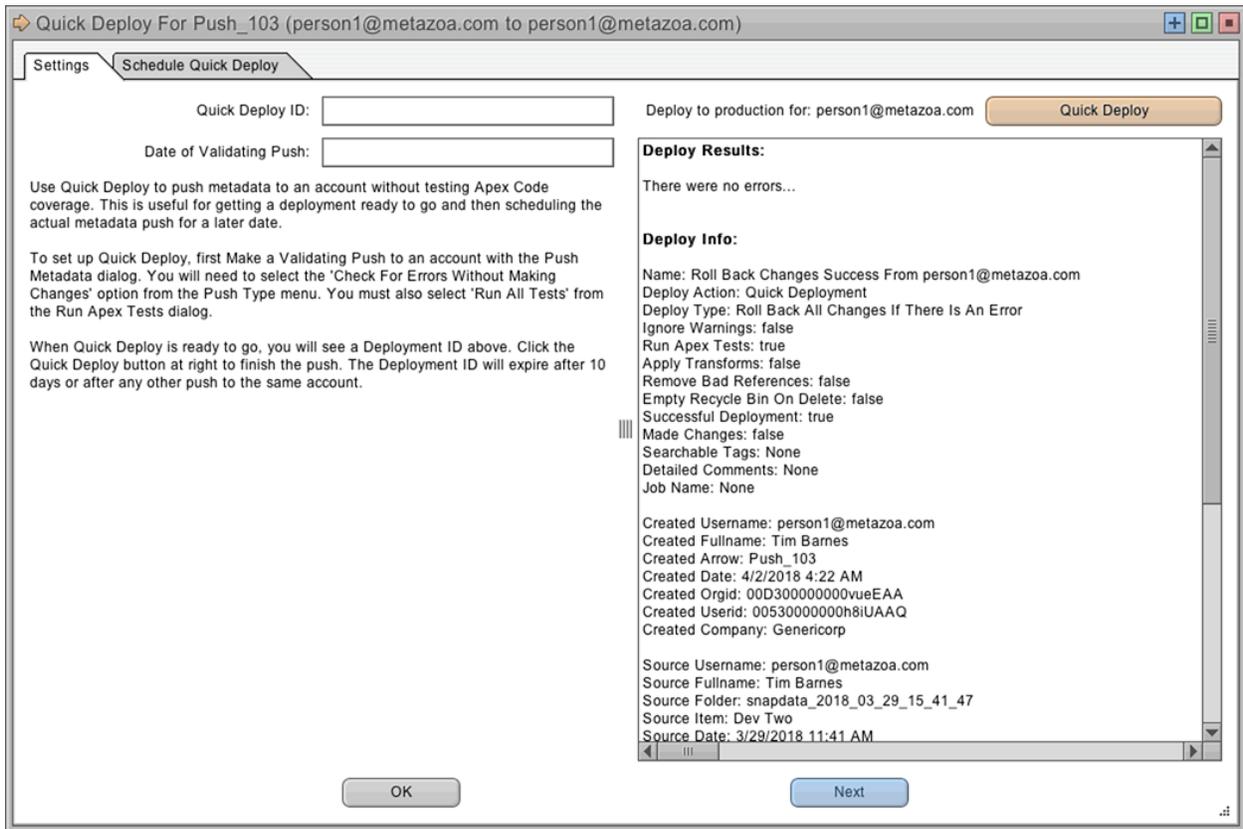
\* This will push all Object fields plus the related settings including: Business Processes, Compact Layouts, Fields, Field Sets, List Views, Record Types, Sharing Reasons, Validation Rules, and Web Links. These settings can also be pushed individually if needed.

\*\* Users tied to workflows must be created in the destination org prior to deployment. Select the Apply Transforms checkbox to automatically change usernames or other values prior to deployment.

\*\*\* This will push all Profile metadata plus the related settings including: Apex Class Accesses, Apex Page Accesses, Application Visibility, Field Permissions, Layout Assignments, Object Permissions, Record Type Visibility, Tab Visibility, User Permissions, and Custom Permissions. These settings can also be pushed individually if needed. If you would like to automatically remove any references to assets that do not exist in the destination or in the Job List from the profiles, select the Remove Bad References checkbox before pushing.

## Quick Deploy

If you conduct a Validating Deployment that runs Unmanaged Tests for code coverage, then you can execute a Quick Deployment instead of the longer Transactional Deployment. A Quick Deployment enables you to schedule deployments to production with more confidence because you know the deployment will succeed and requires a shorter window of time. A Salesforce service update is also less likely to interrupt a Quick Deployment.



Quick Deploy For Push\_103 (person1@metazoa.com to person1@metazoa.com)

Settings | Schedule Quick Deploy

Quick Deploy ID:

Date of Validating Push:

Deploy to production for: person1@metazoa.com **Quick Deploy**

**Deploy Results:**  
There were no errors...

**Deploy Info:**  
Name: Roll Back Changes Success From person1@metazoa.com  
Deploy Action: Quick Deployment  
Deploy Type: Roll Back All Changes If There Is An Error  
Ignore Warnings: false  
Run Apex Tests: true  
Apply Transforms: false  
Remove Bad References: false  
Empty Recycle Bin On Delete: false  
Successful Deployment: true  
Made Changes: false  
Searchable Tags: None  
Detailed Comments: None  
Job Name: None

Created Username: person1@metazoa.com  
Created Fullname: Tim Barnes  
Created Arrow: Push\_103  
Created Date: 4/2/2018 4:22 AM  
Created Orgid: 00D300000000vueEAA  
Created Userid: 00530000000h8iUAAQ  
Created Company: Genericorp

Source Username: person1@metazoa.com  
Source Fullname: Tim Barnes  
Source Folder: snapdata\_2018\_03\_29\_15\_41\_47  
Source Item: Dev Two  
Source Date: 3/29/2018 11:41 AM

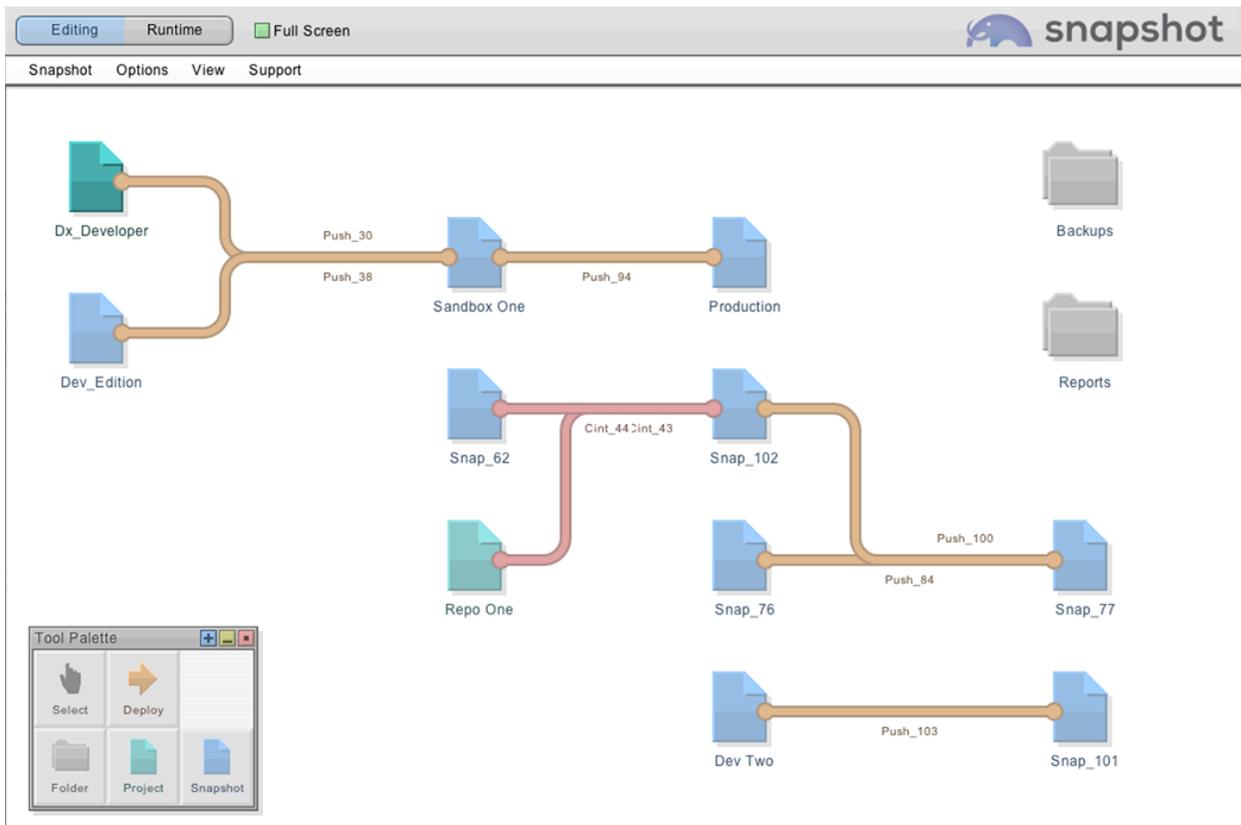
OK Next



## Metadata Flows

Snapshot provides tools to create a visual map of your Salesforce release environment. In the example below, there is a developer project and a Developer Edition org connected to a partial Sandbox, which in turn is connected to a production org. All of the snapshots and metadata deployments on the desktop can be scheduled and automated as needed.

The Snapshot desktop provides a visual guide to metadata flows from the source to destination. Your implementation might be more or less detailed depending on your process and needs. The Snapshot desktop provides a flexible way to rewire your release environment over time and accommodate new projects and reports.



## Compliance Reporting

Snapshot has two dozen reports that are hugely beneficial for corporate governance and security. All of these capabilities can be scheduled and automated. These reports will be the subject of a separate white paper, but here they are listed, below:

### **Comparison Reports**

- Time Series Differences
- Metadata Differences

### **Metadata Reports**

- Asset History
- Similar Assets
- Data Dictionary
- Relationship Matrix
- Fields Vs Page Layouts
- Record Types Vs Picklists
- Controlling Vs Dependent Picklists

### **Data Usage Reports**

- Field Usage
- Picklist Usage
- Last Activity Date
- Apex Code Coverage

### **Security Reports**

- Record Level Security
- User Activity Timeline
- Security Health Check
- Profiles
- Permission Sets
- Combined Security

## Conclusion

This whitepaper has discussed the challenges that companies face with change and release management on the Salesforce platform. The Snapshot product from Metazoa provides a best-of-breed solution to enable effective release management through streamlined deployments, reporting, and compliance.

[support@metazoa.com](mailto:support@metazoa.com)

1-833-METAZOA

1-833-638-2962

<https://www.metazoa.com>

Twitter: @metazoa4sf

Facebook: <https://www.facebook.com/metazoa4sf>

LinkedIn: <https://www.linkedin.com/company/18493594/>