CERNA SOLUTIONS®

How to Prepare for a ServiceNow Upgrade

GUIDANCE DOCUMENTATION

Need help upgrading ServiceNow?

Contact Us:

Email: info@cernasolutions.com
Phone: +1 844 804 6111 (US)
+44 (20) 33254077 (UK)

TABLE OF CONTENTS

BEFORE YOU UPGRADE

WHY UPGRADE?

- ServiceNow supports the most recent release, as well as the immediate previous release.
- ServiceNow only provides patches and hotfixes for supported releases.
- Upgrading to the most current version of ServiceNow allows customers to take full advantage of the newest functionality of the platform.
    - This enables you to utilize **new products** made available by ServiceNow
    - Your instances will be **more secure**, have higher availability, and their performance will be better.

WHEN TO UPGRADE

- Wait until at least one or two major patches have been released before upgrading to the newest release. (Approx. 1 month)
    - Allows ServiceNow time to patch any major issues with the release.
    - Leverage feedback from the ServiceNow Community about outstanding issues.
- If possible, do not get behind any more than one major release. The more releases you wait the larger the process will be and the more issues you will be likely to encounter.

PUT TOGETHER YOUR TEAM FOR THE UPGRADE

TESTING TEAM	TECHNICAL TEAM	PROJECT MANAGER	KEY STAKEHOLDERS
1-10 MEMBERS	1-4 MEMBERS	1 MEMBER	1-4 MEMBERS
Group of individuals that know the business processes and can create test cases that are reusable and documented. This can include power users.	Group of technically skilled individuals who understand ServiceNow	Leads the upgrade endeavor and ensure it stays on track	People in charge of business processes, process managers, subject matter experts. People who have been tapped for ownership of processes of the platform

HOW TO PREPARE

- Before you begin the upgrade process, it is essential to read the release notes for your target version, so you understand the required upgrade and migration tasks. This information helps you plan a safe and effective upgrade. These can be found in the ServiceNow docs.
  - Offer valuable information about new functionality, notable changes, and fixes available in the new release.
  - Help you determine whether items you previously customized are being upgraded.
  - Understand if there are any enhancements implemented previously that are now made available Out of the Box (OOTB) with the new upgrade
  - Understand if a new feature is added to replace an existing custom-built feature
- Read through the ServiceNow community for any notes related to the upgrade that may benefit you.
- Complete the Upgrade Planning Checklist on ServiceNow's Website
  - Orlando Planning Checklist

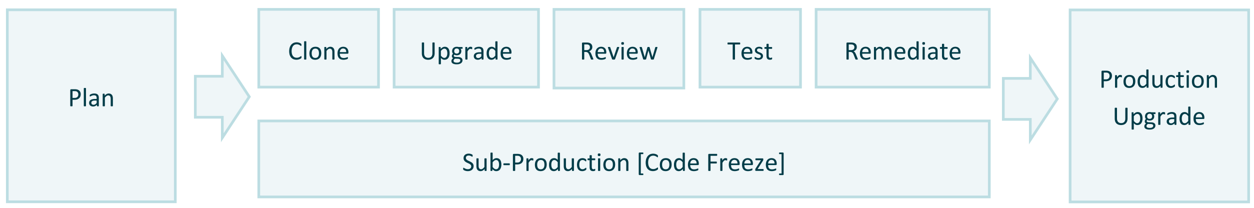
WHAT YOU NEED TO KNOW

COMMON PITFALLS

Why upgrades can be painful	How to protect yourself
➢ Difficult to predict breaking points ➢ Just because you modified an object doesn't mean it will break ➢ Just because you didn't modify an object doesn't mean it won't break ➢ There are no shortcuts to testing (besides automation) ➢ New bug discovery	➢ Get firm requirements for the largest processes ➢ Integrations tested in subprod environments ➢ Clone, test, repeat ➢ Clone the day before of Upgrade to have a backup instance

HOW TO UPGRADE

PROCESS OVERVIEW



CREATE TEST PLAN

- Create a comprehensive test plan that includes test cases for all core instance functionality and integrations, including any customizations you may have.
- Impacted Products and Applications - Create a list of applications and features being used and/or will be used post-upgrade. This will help you understand what exactly you have in your environment that will be impacted during the upgrade and what steps you will need to put into your plan for remediation and/or configuration for each product. This is also be beneficial to understand when allocating resources for testing.
    - Read through the Features and Changes by Product
    - Read through Pre-Upgrade and Post-Upgrade tasks for various products, paying special attention to the products on your list
- Impacted Plugins – Review the Changes to plugins in the New York release and create a remediation and/or configuration plan for each.
- Automated Testing: Make sure that your automated test scripts are up to date

PREPARE FOR UPGRADE: SUB-PRODUCTION INSTANCES

- Implement code freeze in your target instance. Ensure you are testing upgrade and not bug introduced by new development.
- Clone over your sub-production instances before you upgrade them. This allows you to have the most up-to-date data to work with when testing. When choosing which instance to upgrade, ServiceNow suggests the following:
    - Upgrade the instances farthest from production first.
    - Consider using a sandbox instance to do a mock upgrade to reduce risk.
- Ensure that your sub-production instances are setup correctly for upgrading. Specifically, check the **Scheduled Job** named **Upgrade** for the following:
    - Check the Upgrade Scheduled Job to see when the job runs next
    - Trigger type is set to Interval
    - System ID is set to None
    - Trigger type is set to Run at System Startup
- Schedule the upgrade for your sub-production instance.
    - Specify which sub-production instance you are upgrading
    - Specify the time and date
    - Select which version you are upgrading to
        - Check relevant patch notes for the specific patches if new patches and hotfixes come out during your upgrade testing

POST UPGRADE: USING UPGRADE MONITOR

- Use the **Upgrade Monitor** to check the status of your upgrade, evaluate what was processed during the upgrade, and what was skipped during the upgrade of your sub-production instance.
    - Navigate to System Diagnostics > Upgrade Monitor
    - You can monitor the status of the upgrade from the Upgrade Progress Page, this will display automatically if an instance is currently being upgraded.
- Once the upgrade is complete, you can check the **Upgrade Summary Report**, which summarizes what actions were taken, has tools for resolving conflicts due to customizations, and provides information for future upgrade timelines.
- Now that the upgrade has completed, it is time to review and process the **Skipped Records List**.
    - Navigate there from the Upgrade Summary Report by clicking on the *Click here* link in the Skipped box. The system will display the System Upgrades form.
    - From the System Upgrade form, you can navigate to **Review Skipped Records** section

RESOLVING CONFLICTS

- Conflicts are listed by priority based on the type of conflict:

  1 (highest priority): xml content
  2: script or script_plain
  3: html content
  4: sys_ui_form_section, sys_ui_related_list, or sys_choice_set
  5 (lowest priority): other

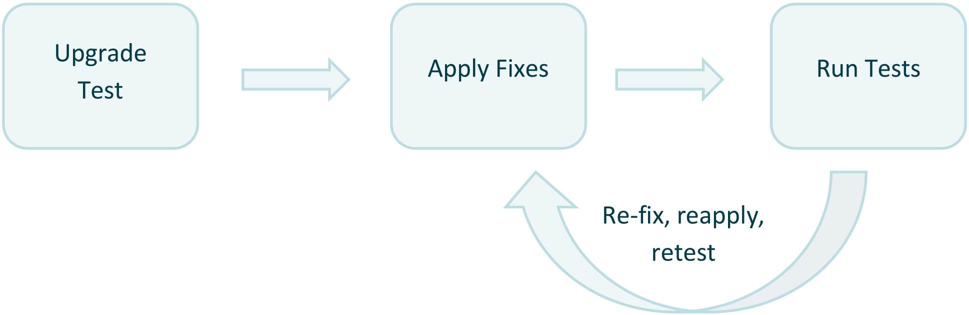
- Evaluate the conflicts for the records in the list and take one of the following actions to remediate:
  - Retain the customized record as is and do not update it
  - Retain the customization by merging changes from the updated object
  - Discard the customization and update the record to match the base system for this upgrade
  - Review the skip and perform no action on the object
  - Leave on the skipped list for a later decision, and note that you have not reviewed the record
- *Notes about resolving conflicts*
  - You don't have to remediate all skipped changes.
  - You should at least acknowledge all skipped changes

RUN YOUR TESTS

- Execute your tests in your dev environment and track the results
  - Compare failed tests against your production copy to confirm that it's truly related to the upgrade
- Fix issues related to any failed tests
- All development work should be captured in update sets
  - Use parent update sets to avoid issues
  - Use proper naming convention and update description to reflect work captured
- Retest to confirm those issues are fixed

TEST YOUR FIXES

- Upgrade your test environment
    - If you don't have a test environment, follow these steps:
        - Export your fixes into a file via export to XML
    - Clone down over sub-production environments.
        - If you have multiple sub-production environments, this should be done to each environment in the promotion path sequentially
        - Otherwise, request a clone of production from ServiceNow and use that
- Apply your fixes in the sub-production environment
    - Import your XML file(s) via update sets
    - Preview update sets for conflicts, resolve them, and apply your fixes
    - Import any other data that you exported out in previous steps
- Run tests – rerun the test cases that were previously established to verify correct functionality
- Re-fix, reapply, retest (LOOP)
    - Re-fix – if a test case does not pass, the developer will need to create a fix in your development instance to remediate the issue.
        - Be sure to capture the work in an update set
    - Run back through the process of migrating your update sets back up to your sub-production clone
    - Retest your test cases in question to ensure that the fix was created and applied correctly



UPGRADE PRODUCTION ENVIRONMENT

- Schedule your upgrade timing – A timeline will need to be created to ensure that the upgrade is following a specific schedule. This should be created by upper management and the identified key stakeholders.
    - Upgrading over the weekend is a common practice, since users will not be using the platform as much during this time.
- Communicate to the affected members – Communicating how and when this will take place is fundamental to your upgrade success. Your communication should include the following:
    - When the code-freeze will happen and who will be affected
        - This will likely be a separate communication as this will not follow the same timeline as the upgrade itself
    - When expected downtime for the production environment will take place
        - As production will be unavailable to the users, this is the most important part
        - Make sure to note how long the expected maintenance window will be
- Apply your fixes to the production environment. Follow these steps:
    - Import your XML file(s) via update sets
    - Preview update sets for conflicts, resolve them, and apply your fixes
    - Import any other data that you exported out in previous steps

POST UPGRADE ACTIVITIES

- All test cases should be reaffirmed post-upgrade and updated accordingly
    - These should be documented via versioning
    - Potential KB articles updated
- Hypercare schedule – this will usually last for a week and potentially up to two weeks
    - This is a time when users can communicate break/fixes to get any outstanding issues resolved quickly
    - Key stakeholders will need to be involved to help communicate issues and set prioritization for any open issues
        - This can be a specific queue that the support team leverages post-upgrade
        - A conference bridge is generally used during the smoke testing post-upgrade
        - Be sure to document everything – this cannot be stressed enough
- Post upgrade communication – once the main issues are resolved the key stakeholders should communicate a successful upgrade to the organization
    - Resume business as usual
    - Remove the code freeze
    - Celebrate a job well done!

Need help upgrading ServiceNow?

Contact Us:

Email: info@cernasolutions.com
Phone: +1 844 804 6111 (US)
+44 (20) 33254077 (UK)