



## UXsyncNow

UXstorm is an independent software vendor that specializes in applications on the ServiceNow platform. We have several applications on the [ServiceNow Store](#) and continually developing new ones. In all our development, we do a lot of development with ServiceNow scripts/HTML/XML. This has led us to creating this tool. We hope it's as useful to you as it is to us. If you wish to support us, check out our User Experience applications on the Store and spread the word of our works! If you are going to the ServiceNow Knowledge event, we will have a booth there so stop on by.

## Brief History

Development in ServiceNow Studio allows for the creation of tables, forms, and many scripts and HTML/XML. While Studio is great for many things, the code editor lacks from a developers perspective as it requires a button press to go full screen and when it saves, it reverts back to a forms view. To continue editing, you must click a button to go full screen and search for the line you were working on. This makes it time consuming to do a simple save and test. In addition, the version control is very rudimentary. This makes it difficult for multiple developers coding or releasing on a singular code base.

To solve these issues, UXstorm has used sn-filesync. This is a great tool but from our use we found the following issues to limit our use:

- Requires editing config files and many of them for different releases/product.
  - Handling conflicts must be done by hand
  - May require clearing changes in the cache
- Special table changes have to be added by hand.
- Doesn't handle conflicts
- Can't switch between instances in the same project
- Designed for global development and not Applications
- Can't be used in automated builds and pushing to our Integration Instance

To solve these issues, we've developed an internal tool, **UXsyncNow**

## Features

- Provides two way syncing between a local work area and a ServiceNow instance
  - Changes to tracked files on the local work area is pushed into the instance
  - Changes to tracked files on the instance are pulled into the work area



- Autodetects all "*files*" on the ServiceNow instance that can be managed
  - Can easily resolve conflicts
    - Pull from Instance and overwrite local file
    - Push local file to the Instance
    - Merge by pulling file from instance and saving it in the local work area with the same name plus extension `.merge`. You can then merge with your favorite merge tool
  - Can resolve one or all conflicts in one command
- Works with ServiceNow Applications
  - Note: You can create a ServiceNow Application in the Global Scope if you need your work to be in global.
- Provides an application command line interface for easy interaction and setup
- Easy setup
- Can support many different configurations in the project to connect to different instances
  - Defaults to a dev config
  - Command line option `-p` (or `--prod`) to use production configuration
- Can run without watching the work-area to support pull or pushing
  - Pull all files from Instance and overwrite create local work area
  - Push the files from the local work area to the instance. We uses this for production builds to our Integration Instance
- Can override a *file* from the Instance to be stored/watched in another location. This is useful for specifying files from a build/generate area to the appropriate ServiceNow file. We use this for our WebPack bundles to UI Scripts.
- Supports using a Proxy Server # Initial Setup

## Installation

Install from NPM

```
npm install uxsyncnow -g
```

Download the UpdateSet from [ServiceNow Share](#). Load the update set into your instance.

**NOTE:** This application is designed for development and should only be used in Development and Test, do not use this on a production instance.

Once the UpdateSet is loaded, any users that you want to use **UXsyncNow** must have two roles assigned -- *admin* and *uxsyncnow\_user\_*. If the user used to connect to the instance doesn't have these roles, the application will not connect.

The updateset only adds a webservice and roles.



## Setup initial Work Area

Goto the directory where you want the work area to be created and follow the following steps: \* Execute uxsyncnow \* Set missing options \* host - set option host uxstormdev1.service-now.com \* user - set option user reed.owens \* password - set option password and enter the password in via the keyboard \* Connect to the instance -- connect \* Retrieve all the applications and tables from the instance - refresh all \* List all the applications - list apps \* Specify the application to sync/watch - set app 14 \* Start the sync/watching - sync

Here is the command output of an initial connection using

Option	Value
host	uxstormdev1.service-now.com
user	reed.owens
password	not shown

```
%uxsyncnow
You are not connected to your instance. Please setup your connection.
host is not defined
user is not defined
password is not defined
uxsyncnow: set option host uxstormdev1.service-now.com
uxsyncnow: set option user reed.owens
uxsyncnow: set option password
The ServiceNow user password to use to connect to the Instance
password = *****
uxsyncnow: connect
Connected!
uxsyncnow: refresh all
Refreshing applications
Refreshing tables
Done.
uxsyncnow: list apps
1) x_uxs_expreport01 Expense Reports version: 1.0.0
2) x_testapp_for_uxsh TestApp for UXsheets version: 1.0.0
3) x_demo_sample Demo Sample version: 1.0.0
4) global [WS] Xplore: Developer Toolkit version: 4.4
5) global Sample Global Application version: 1.0.0
6) global UXstorm's UXsyncNow version: 1.0.0
7) x_uxs_ls UXstorm LocalScript version: 1.1.17
8) x_uxs_demo_printer UXstorm Printable CR Demo version: 1.0.0
9) x_invoice Invoice version: 1.0.0
10) x_193105_invoice Invoice Demo version: 2.0.2
11) x_uxs_leave Leave Request version: 1.0.3
12) x_uxs_pdf UXstorm PDF version: 1.3.13
13) x_uxs_uxsheets UXsheets version: 0.1.12
14) x_test_app_for_uxs Test App for UXsyncNow version: 1.0.0
uxsyncnow: set app 14
lookup :14
Found : Test App for UXsyncNow (f58f6f7df793030022d7e4c7238dff47)
uxsyncnow: sync
uxsyncnow:
```



This initialization will create a file in the current directory named `uxsyncnow-config-XXX.json` and a directory named `.uxsyncnow-XXX` where `XXX` is the configuration type. By default this will be `dev`. If ran with a `-p` option then it will be `prod`. The `.json` file should be put under source control and the directory ignored by source control. The directory contains the cache information.

## Running with existing Work Area

If a work area has already been setup, then all you need to do is run **uxsyncnow**. If the work area is the dev work area, nothing extra needs to be specified. You can specify the production work area by using the command line argument `-p` **uxsyncnow** `-p`. Additional work areas can be used by specifying their name with the `-c` option. For example a work area called `reed` can be specified by **uxsyncnow** `-c reed`.

## Command Line Options

**uxsyncnow** supports both short and long format for the command line options. Short options are one char and are preceded by a single `-`. Long options are words and preceded by two `-`

Option	Description
<code>-a areas, --areas areas</code>	Specifies the areas in the application that the debug statements will be printed. <i>areas</i> is a comma decimated list of the areas. <i>See debug areas Application Command for list of areas</i>
<code>-c type, --config type</code>	Specifies the type of configuration to use by <i>type</i>
<code>-d num, --debug num</code>	Sets the debug level to <i>num</i>
<code>-h, --help</code>	Prints out help information
<code>-p</code>	Specifies to to use the production configuration
<code>--pull</code>	Uses PULL for the sync method
<code>--push</code>	Uses PUSH for the sync method
<code>--sync</code>	Uses SYNC for the sync method. This is the default method
<code>--nowatch</code>	Causes the application only to process the synchronization of the work area and doesn't watch for changes on the instance or the work area. When the synchronization is done, then the application will exit



## Application Command Line

**uxsyncnow** provides an application command line interface to interact with the application. When the application is started you will receive a command prompt `uxsyncnow:` and can run any of the application commands via this interface. Try `help` to get started.

### conflicts

Lists all the conflicts detected by sync

### connect

Attempts a connection to the ServiceNow instance. If there are any issues in the connection, appropriate error messages will be provided.

### debug areas

Shows all areas of **uxsyncnow** that can be filtered by *debug filter <area>*.

### debug filter <area>

This sets an area of **uxsyncnow** that will be shown if debug level is  $> 0$ . When **uxsyncnow** starts, there is no area so all areas are shown. Once any area is specified, only those areas will be shown in the debug output.

### debug level <level>

Sets the debug level. A debug level of 0 effectively turns off debugging output. Most areas use debug level 1.

### debug reset

Resets debug level to 0 (off) and clears any debug areas that were set.

### exit

Exits **uxsyncnow** CLI.



## help [command]

Provides help on the **uxsyncnow** CLI commands. Specific help can be obtained by specifying it on the help command like `help set`.

## list apps

Lists all applications on the instance and their corresponding ID that can be used for setting the application. See *set app num*

## list files

List all files mapped from the in the work area.

## list options

List all the options, help on the options, and their current value.

## list overrides

List all the overrides for the work area

## list tables

Lists all the tables that contain any field that is a JavaScript, HTML, or XML type and the record is tracked in update sets.

## override add <source> <dest>

Adds an override to the local work area for a default application file mapping. By default each file is placed in a directory named as the table name in ServiceNow. File names in the work area are in the pattern of `<name_of_record><fieldname>.[js/html/xml]`. If you need the file to be named differently or be in another directory or even both, then you need to do an override.

`<source>` must be the relative full path from the project top directory (where **uxsyncnow** was ran). You can find all the file by CLI command `list files`.

`<dest>` is a relative path and filename from the project top directory. If it starts with `/` or `\` then it will be treated as an absolute path from the root of the filesystem.



## override remove <source>

Removes the override for <source>

## refresh apps

Refreshes the list of Applications from the instance. Do this to pick up any new Applications added to the instance or to get the initial list of applications.

## refresh tables

Refresh the tables from the Instance. Do this to get the initial list of tables from the instance or if any new table is added to the instance that is tracked in update sets and contain js/HTML/XML fields that need to be tracked.

## refresh all

Refreshes both the Applications and the Tables from the instance.

## resolve <conflictNumber> <mode>

Resolves a conflict specified by <conflictNumber>. All conflicts can be resolved by using *all* for the <conflictNumber>.

<mode> specifies the mode that is used to resolve the conflict. Possible choices are PULL, PUSH, or MERGE.

### PULL

This will pull the file from the instance and overwrite the local work area file with it's contents.

### PUSH

This will push the contents of the local work area file to the instance.

### MERGE

This will pull the file from the instance and save it's contents in a file in the local work area. The filename will be the same as the work area filename with an extension of .merge.

For instance if the local file is in sys\_scripts/my.js then the new saved file will be sys\_scripts/my.js.merge.





Use this mode to use your favorite merge tool to resolve the conflicts.

If you use an IDE such as Webstorm, you don't need to use this mode. You should use PULL and have Webstorm detect the change and do the merge with changed local file.

## set

The *set* command is a command group as it is used to set values in multiple areas. ### set app *num* Sets the application to be synchronized. Find *num* by running `list apps` to find the application number for each application. After specifying a new application you must run `sync` to perform the synchronization/watching of the new application.

### set option <option> [<value>]

Sets the specified option <option> to the provided value. If <value> is not provided, then you will be prompted to enter the value.

NOTE: It is recommended to use the prompt for the password option as the password is not shown.

Valid options are:

Option	Description
connection_max	Maximum number of connections to the ServiceNow instance. If you experience problems with too many connections then you can change to number something low like 5 or less and see if it fixes your experience. A value of 0 defaults to the system's default number of connections.
connection_wait	Number of milli seconds to wait between requests. This is used not to throttle the instance with too many requests at a time. A value of 0 defaults to the systems' default wait time.
host	The host name of the ServiceNow Instance
interval	How many milli seconds to wait between checking the instance for any changes. The default is 30000 or 30 seconds. You can not set this to value less than 1 second (1000).
password	Password of the user to connect to the ServiceNow instance. NOTE: This value is encrypted and you can never see it's value.





port	Port to use to connect to the ServiceNow instance. A value of 0 specifies to use the default port.
protocol	The web protocol to use for the ServiceNow instance. This can be either http or https. The default is https
proxy	Specifies a Web Proxy server to send all requests through.
top_dir	By default, all the directories of files from the instance will be created in the current directory where <b>uxsyncnow</b> is ran. If you want to have the instance directories and files in a subdirectory, you can do this by specifying the top_dir. This will create a directory as specified and will contain work area._
user	The username of the ServiceNow user to connect to the instance. NOTE: This user must have the roles of <b>admin</b> and <b>uxsyncnow_user</b>

## sync

Synchronizes the local work area and the Application files on the instance. It uses the current synchronization mode. NOTE: This only needs to be used during first initialization or if you make any configuration changes requiring a re-sync. By default, **uxsyncnow** starts up in sync mode if it can connect to the instance and an application was set.