Jared Gerhardt

Unit Operations Guide

Swing SP Configuration

Contents

[General Outline 2](#_Toc180411683)

[Application Set-Up 2](#_Toc180411684)

[General Considerations 3](#_Toc180411685)

[Measurement Thread 4](#_Toc180411686)

[Specific Functions 5](#_Toc180411687)

[Powder Dispense with GDU-Pfd 5](#_Toc180411688)

[Liquid Dispense with GDU-V (Volume Based) 6](#_Toc180411689)

[Liquid Dispense with GDU-V (Mass Based) 6](#_Toc180411690)

[Volumetric Dispense with Four-Needle Head 7](#_Toc180411691)

[Heating/Stirring with the Tumble Stirrer or Shaker 7](#_Toc180411692)

[Heating with the Tumble Stirrer or Shaker 7](#_Toc180411693)

[Stirring with the Tumble Stirrer or Shaker 8](#_Toc180411694)

[Filtration with the SPE Rack 8](#_Toc180411695)

[Overview 8](#_Toc180411696)

[Solvent Washing Function 8](#_Toc180411697)

[Separate Filtration Functions 10](#_Toc180411698)

[Transport with Gripper MTP 10](#_Toc180411699)

[Pauses and OK-Dialogues 10](#_Toc180411700)

[Data Logging and Exporting CSV Files 10](#_Toc180411701)

# General Outline

## Application Set-Up

Step 0

Create a new folder in 'SwissCat Github > Chemspeed\_Autosuite\_programs > UO\_CSVs' that is for your specific reaction on a certain date.

From the folder 'SwissCat Github > Chemspeed\_Autosuite\_programs > UO\_General Templates', copy and paste the 'Excel' folder, 'CSVs' folder, and 'App' folder into the new folder that you just created.

Step 1

From the general UO\_SwingSP App, create the workflow for your reaction. See figure xxx for an example.

\*As of Oct. 21st, transport tasks and pauses (e.g. to put on paradox cover) must be inputted manually, as seen in figure 1

\*Set variable Operation\_Counter to the correct integer for each step. For each function input, put 'Operation\_Counter' without the apostrophes.

Step 2

Fill out the excel file for each unitary step accordingly. There are specific instructions in each file. When naming the file, replace "0" with the number of it's corresponding workflow step.

Example: If the third step is a 'LiquidDispense\_GDU-VVolume', the excel file should be named '3LiquidDispense\_GDU-VVolume'

VERY IMPORTANT: Only change the number of each excel file name. If you change anything else, the Autosuite app will not import that CSV file.

Step 3

Once all excel files are filled out, you must save them all as CSV files. Save them into the folder 'CSVs > Imports' within the folder that you created in Step 0.

VERY IMPORTANT: There are multiple CSV file types when saving from excel. Save the file as 'CSV (Comma delimited)'

Step 4

If the above steps were completed on your own computer, download a local copy of the new folder you created in Step 0 from github onto the SWING SP laboratory computer.

Create a local copy of the most current UO\_EPAGNY\_Swing SP App, and place it into the App folder within that local copy.

Step 5

In the global variables, you will need to change the 'CSV\_Import\_Filepath' and the 'CSV\_Export\_Filepath'. To do this, open up the CSVs > import folder, and copy and paste the filepath into the 'CSV\_Import\_Filepath' initial value. Next, open up the CSVs > Export folder, and copy and paste the filepath into the 'CSV\_Export\_Filepath'.

Step 6

Run the reaction in a simulation to ensure that the import/export filepaths are correct.

VERY IMPORTANT: An import failure will NOT show up as an error message. You must go into the log > custom messages. If you see any failure to import/export CSV file, then doublecheck the filepaths, CSV file names, and Operation\_Counter Integers.

## General Considerations

* Specific instructions for each unit operation are written in the template files
* Zone/Element names in the CSV files must match exactly with the zone/element names in the AutoSuite App
* Anything written in the CSV file after the ‘End’ row will not be imported into AutoSuite, i.e. you may write notes to the user below the ‘end’ line
* As of Sep. 19th, the SPE Filtration task is not fully optimal – will update later
* Boolean Variables: For these variables, input 1 for true and 0 for false
* Do not open/edit any import or export CSV files while the application is running and trying to access that file. If you do, the system will give an error and the platform will have to be manually reset.

A screenshot of a computer

Description automatically generated

Figure - Example of Complete Reaction Workflow in AutoSuite

## Measurement Thread

Here is a summary of how the measurement thread is carried out in the Swing SP. This is included for you to have a comprehensive understanding of how it works, in case you would like to change it for your specific reaction.

Firstly, it executes while the variable ‘Reaction\_Running’ is true. It is set to true at the beginning of the reaction, and is set to false in the reaction thread after the workflow is completed. Ensure the ‘Set Variable’ task that sets Rection\_Running = false is located within the reaction thread. If it is not, the measurement thread will keep looping until you manually stop it.

In the Swing SP, three variables can be measured: temperature, stirring/shaking speed, and pressure. Certain parameters are recorded for each variable, and if they vary by a certain margin from the previous value after 10 seconds, the change will be recorded in the export file, along with the absolute time when that change occurred and the time since the start of the application (in seconds). If there are no changes after 10 minutes, the code will automatically export the current conditions in order to maintain a constant profile in the CSV export file.

The following parameters are recorded for the temperature.

* Temperature of the shaker [°C]
* Temperature setpoint of the shaker [°C]
* Temperature of the tumble stirrer [°C]
* Temperature setpoint of the tumble stirrer [°C]

If any of these parameters change by +/- 1 °C from their previous value after 10 seconds, that change will be recorded in the export file.

The following parameters are recorded for shaking/stirring.

* Rotation of the shaker [rpm]
* Rotation setpoint of the shaker [rpm]
* Rotation of the tumble stirrer [rpm]
* Rotation setpoint of the tumble stirrer [rpm]

If any of these parameters change by +/- 2 rpm from their previous value after 10 seconds, that change will be recorded in the export file.

The pressure measurements are more important in the Catscreen platform, since there are no pressure blocks/Isynth racks, or other pressurised apparatus in the Swing SP platform. As of 18/10/24, I have not found a way to access the current pressure of the whole platform. The access path given in the SwingXL manual is not working. I will try to find away around this.

# Specific Functions

## Powder Dispense with GDU-Pfd

Function name: PowderDispense\_GDUPfd

* This is for dispensing powders using the GDU-Pfd
* The title of the CSV file should be the operation counter + ‘PowderDispense\_GDUPfd’
  + Example: ‘2PowderDispense\_GDUPfd’ if it is the second task in your workflow
* Destination Zone: This can be a series of wells or a single well. However, it must match a zone name exactly in the AutoSuite App.
* Source Zone: This will be one of the GDU-Pfd Containers, named ‘A1\_PfdContainer’, ‘A2\_PfdContainer’, etc… with A being the first row and B being the second row.
* Target Amount: This is the target amount of powder you want to dispense, in mg.
* Tolerance: This will be the tolerance used by AutoSuite when dispensing. I have found that around a 0.3-0.5mg tolerance is okay for most powders. However, the best tolerance to use will change depending on the grain size and physical properties of the powder. Too low of a tolerance might cause the AutoSuite app to ‘crash’ since the GDU-Pfd task will stall out when it can’t dispense within the tolerance you want.
* Autodispense: See advice on Boolean variables. If you want to use autodispense, input 1, and you do not need to include any more parameters in the CSV file besides the Solid Name. If you do not want to use autodispense, input 0 and you must fill out the rest of the parameters. See the Swing XL manual for the acceptable ranges of these varaibles.
* Solid Name: Input the solid name of the substance you are dispensing. This will not be used by Auotsuite itself, but it will be exported into the export CSV file to make the data analysis easier.
* As always, put ‘End’ after the final row.

## Liquid Dispense with GDU-V (Volume Based)

Function name: LiquidDispense\_GDUVVolume

* This function is for dispensing liquids with the GDU-V using a target amount that’s volume based.
* Destination/Source Zone IDs: Same as other functions. However, it is better for this task to use destination zones that are multiple wells if you can, instead of doing it well-by-well. This is because the app will automatically thresh the tip after each row. If you want to change this, you must change it in the Application
* Tip Zone ID: This is the tip you will use for that specific dispense. Ensure the name matches the tip zones in AutoSuite.
* Target Amount: This is the target amount you want to dispense in ul.
* Aspiration Height: This is the height difference from the top of the vial that you are aspirating from in mm. Advice for the values in the SwingXL platform is given below.
  + When aspirating from MTP vials (24x4ml) in the SwingXL configuration, -46mm reaches the bottom of the vial. Note: for MTP vials, do not use the 2.5ml, since the tip becomes to thick to reach all the way down. It will end up causing damage to the tip and/or the vial.
  + When aspirating from the stock solution vials (11x20ml) in the SwingXL configuration, -49mm almost touches the bottom of the vial.
* Autodispense: Boolean variable to set if you want autodispense on or off.
* Speed Mode/High Stability Mode: If you choose autodispense, you can choose one of three modes. The exact mode you should choose depends on the liquid and how accurate/quickly you want the dispense to be.
  + Speed mode ON (1) and High Stability OFF (0): fastest, least accurate
  + Speed mode OFF (0) and High Stability ON (1): slowest, most accurate
  + Speed mode OFF (0) and High Stability OFF (0): Speed between the first two, decently accurate.
* Dispense Height: This is the height difference from the top of the vial when you’re dispensing in mm. If you want accurate mass feedback, it is beneficial to set this to around 1 or 2mm, since the mass feedback will be messed up if the tip touches the side of the vial when dispensing.
* Liquid Name: This is the name of the liquid/solution you’re dispensing. This will not be used by Autosuite, but it will be exported into the export CSV file.
* Equilibration Time: This is the amount of time in seconds that the tip will wait after dispensing in order to let any droplets that might still be on the tip fall. The default is 0 seconds, but a longer time should be considered for more viscous liquids.

## Liquid Dispense with GDU-V (Mass Based)

Function name: LiquidDispense\_GDUVMass

* Largely the same to the GDUVVolume function, with only some differences
* The target amount should be inputted in mg
* After the column for the liquid name, there are the parameters if you set autodispense equal to false. There must be values for these if autodispense is not chosen.

## Volumetric Dispense with Four-Needle Head

Function name: VolumetricDispense\_4NH

* This function is for dispensing or transferring liquids with the 4NH
* As of now, only the first needle head (with a volume of 1ml) is used for this task. If you want to change this, you will need to change the needle selection in AutoSuite
* The destination zone ID is any well or group of wells defined as a zone in AutoSuite
* The source Zone ID can be a valve port or a separate vial if you’re transferring a liquid. If you are using one of the valve ports, type ‘valve ports B’ or ‘valve ports C’ (without the apostrophes) depending on which valve port you want to use. Alternatively you can write the name of a zone defined in AutoSuite
* The target amount is in mL
* Rinsing is a boolean variable: 0 for no and 1 for yes
* If rinsing is selected, choose a valve port for the rinsing solvent. Type ‘valve ports B’ or ‘valve ports C’ (without the apostrophes) for which one you want.
* The aspiration height is the height difference (bottom up) in mm that the 4NH aspirate at. Leave this blank or set it to 0 if aspirating from a valve port.
* The dispense height is the height difference (top down) in mm that the 4NH will dispense into the destination well at.
* Enter the liquid name of the liquid that is being transferred/dispensed. This will not be used by AutoSuite for the task, but it will be utilized in the export file.

## Heating/Stirring with the Tumble Stirrer or Shaker

Function name: HeatingAndStirring\_TumbleStirrerShaker

* This function is for when you want to heat and shake/stir a rack at the same time
* In the SwingSP platform, the heating zone ID can either be ‘Shaker’ or ‘Tumble Stirrer’ (without the apostrophes)
* The heating temperature is in C
* The temperature gradient in C/min is how quickly the tool chosen should heat up. If left empty, it will heat up as quickly as possible
* The stirring speed is in rpm
* The heating and stirring time is in min

## Heating with the Tumble Stirrer or Shaker

Function name: Heating\_TumbleStirrerShaker

## Stirring with the Tumble Stirrer or Shaker

Function name: Stirring\_TumbleStirrerShaker

## Filtration with the SPE Rack

### Overview

Function name: Filtration\_SPE

This function can be broken down into a series of volumetric dispenses and transfers depending on your specific workflow and needs. It was created in order to simplify the filtration coding, but you could simply use separate volumetric dispense tasks using the 4NH.

There are three different types of zones in the SPE rack: collect, direct, and waste. For clarity, these are explained below.

* Collect: When dispensing into a collect SPE zone, the solid/liquid mixture will enter the filtration tip and the liquid will drip down into the SPE vials. You can not aspirate from the collect zone, only dispense.
* Direct: The direct SPE zones allow the 4-Needle head to reach into the vials located in the bottom half of the SPE rack. You can aspirate from the direct zones to collect the filtrated liquid and dispense into a different rack.
* Waste: When dispensing into waste zones, the liquid will go to waste. You cannot aspirate from this zone, only dispense.

The filtration\_SPE function was used specifically for transferring from the paradox rack 🡪 SPE Collect and then from the SPE Direct 🡪 HPLC Rack. However, it would be better (and more adaptable) to simply use two 4NH transfer functions if that’s all you require. For example, the first task would be a volumetric transfer from the reaction plate to the SPE Collect zone. Then, a wait of 10 minutes or so would be added to allow the solution to fully filtrate. Finally, a second volumetric task would be used to go from the SPE direct zone to the final vials.

However, it is usually beneficial to have some sort of solvent washing to ensure that all of the product is transferred out of the vial. In this case, after the first volumetric transfer from the reaction rack to the SPE Collect zone, you would fill each vial with a solvent, shake/stir, and transfer that to the collect zone as well. This would help collect any product that is still stuck to the side of the vials. To carry this out, the function ‘SolventWashing’ can be used, or you can simply use a series of volumetric transfers, transports, and shaking tasks to achieve the same goal. Both methods are described in detail below.

### Solvent Washing Function

The solvent washing function is useful in many circumstances, but may be too limited for your specific reaction. However, it is outlined here in case it is usable for your situation.

* The Destination Zone ID is the zone of the reaction rack. In other words, it is the vials that should be ‘washed’ with solvent.
* The Source Zone ID is the zone of the solvent used for the washing step. As before, if it is a valve port, type ‘valve ports X’, with X being the specific valve port. If it is a vial or multiple vials, ensure the zone matches exactly with AutoSuite.
* The SPE Collect Zone ID is the zone of the SPE Collect filtration tips/vials you will be dispensing into. This can be multiple wells or a single well. However, ensure that the total number of wells into this zone match the total number in the destination zone ID.
* The washing amount is the amount of solvent in mL added to each vial per wash. If you had an effective reaction volume of 1.2mL, then it is recommended to add around that amount to achieve an affective wash.
* The wash times are the total number of washes you do.
* The solvent name will not be used by Autosuite in the task, but it will be used in the export CSV file.
* ‘Shaking’ is a Boolean variable. It will set if the rack should be shaken before the solvent added will be aspirated and dispensed in the SPE rack. Write 0 for no, and 1 for yes. It is recommended that you include this step, as it will ensure a much more effective washing step.
* The shaking speed is in rpm.
* The shaking time is in minutes.
* The paradox rack footprint is the location on the paradox rack that the reaction rack should go. You can choose 1, 2, or 3. 1 will be closest to the front of the machine, and 3 will be in the back. See the configuration on Autosuite for an illustration of this.
* The shaker rack footprint is the location on the individual shaker rack that the reaction rack will be placed. You can choose 0, 1, or 2. 0 is closest to the front, and 2 is in the back.
* Amount to Collect is the volume in mL that the 4NH will transfer from the reaction rack to the SPE rack each wash time. It is recommended that this value is slightly higher than the amount of solvent added per wash, in order to ensure that all of the solvent is aspirated and filtrated.
* Rinse to Collect is a Boolean variable to set whether you want the 4NH to rinse the needle after each aspiration to the collect zone.
* The rinse port is the port that the machine will rinse from. Type ‘B’ or ‘C’ without the apostrophes.

This function was made with one specific reaction in mind, so the limitations are listed below.

* The washing amount is constant through each wash.
* Only works with the individual shaker, not the tumble stirrer.
* The reaction rack must be located on the paradox rack to start.
* The transport tasks that move the rack between the paradox rack and the shaker rack is hard-coded. If you need to change certain parameters like the grip depth/force, you must do that manually in the App.
* This task must start with vials that have a low amount of liquid, i.e. a separate volumetric transfer task (reaction rack 🡪 SPE collect) needs to be included before this step.

### Separate Filtration Functions

Alternatively, the SPE filtration step can be split up into it’s component parts the system requires more flexibility or different constraints. An outline of what that would look like is shown below.

1. Initial transfer to SPE Rack
   1. Volumetric transfer with 4NH from reaction vials to SPE Collect zone
2. Solvent washing
   1. Volumetric transfer of solvent to the reaction vials.
   2. Transport reaction rack to the individual shaker or tumble stirrer.
   3. Shaking/Stirring for a certain amount of time.
   4. Transport reaction back to the initial location.
   5. Volumetric transfer of solvent to the SPE Collect Zone.
   6. Loop steps 2.a-2.e however many times is necessary.
3. Transfer to final vials
   1. Volumetric transfer with 4NH from SPE Direct Zone to final vials (ex: HPLC rack)

If this method was chosen, you would simply use the corresponding unit operations for each step.

## Transport with Gripper MTP

As of 18/10/2024, the transport function is not working in Autosuite. Use a manual transport task each time you need to transport a rack.

## Pauses and OK-Dialogues

Enter manually any pauses or OK-dialogues that are necessary for the reaction.

# Data Logging and Exporting CSV Files