Execution of this Actions Depends on the Following Action(s)

This field on the Admin tab of Actions allows you to specify no, one or multiple alias(es) of other Actions.
Before an Action is effectively executed, any Actions with the alias(es) entered into this field must run first.

⚠ **NOTE that as opposed to references that can reference the result of an Action, which has already run, across Runtypes, dependencies require the Action being dependent on to run in the same RunType, too.**
In essence, dependencies are comparable to references with an execution type of run always (!).

Good Example 1:
Action A and B both run after login, A depends on B (that is, the alias for Action B is entered into the dependencies field of Action A)
If B has already run, A will run.
If B has not run yet, A will attempt to run B. As soon as B has finished, A runs.

Bad Example 1:
Action A runs after login, B before login. A depends on B.
Irregardless of whether B has already run by the time A wants to run, A will fail to execute as B is not configured for the same Runtypes.

Bad Example 2:
Action A runs after login, B at shutdown. A depends on B.
Since B has not yet run by the time A is set to execute, and B is configured for a later Runtype, execution of A fails.

⚠ **Execution of an Action can NOT be brought forward across Runtypes through references or dependencies.**

Bad Example 3:
Action A depends on B, however, a user does not see Action B.
Action A will therefore not run as it cannot resolve the dependency.

Bad Example 4:
Action A and B both run after login, A depends on B.
B is tied to a Conditions preventing it from running.
Since A cannot make B run, A will not run either.

Bad Example 5:
Action A depends on B, and B again on A.
This is a circular reference - neither A will run, nor will B.

⚠ **Any Actions that are part of a circular reference will be excluded from execution entirely.**

⚠ **NOTE that you can have nested dependencies, e.g. A depends on B depends on C depends on D.**
Similarly A could depend on B1, B2 and B3 and B3 again on D1 and D2. In this example, the order of execution (independent of whether an Action already ran) would be B1, B2, D1, D2, B3, A.
If executing any action in this example chain fails, the remaining Actions will naturally not be executed. Any already executed Actions will not be rolled back.