In Jira, issue links have no effect over the environment's structure and behaviour. That means we can create any type of dependency between them in various combinations. Even a type of dependency that would make no logical sense or would end up in a paradox can exist in the vanilla Jira environment.

In our application though, links have a direct effect on the Start and End Dates, so some combinations of links will not be allowed. This concerns what we call "loops". Let's imagine the following type of inter-dependency:

Task A (Ends) (Starts) Task B
Task B (Ends) (Starts) Task A

Such a combination of links on these 2 tasks would result in Task A rescheduling Task B, Task B rescheduling Task A, Task A rescheduling Task B, and so on and so forth.

This sort of paradox would create an infinite number of data re-calculation and could possibly crush our environment. This is why our app will not allow it and will return a Warning.

Loops are very easy to detect if tasks belong to a single Program. However, at times we may meet a situation when the same task is a component of various - multiple Programs, each belonging to a different user. We could cause a change that might reschedule their whole Gantt, sometimes even not realizing it (i.e. once permissions to their Program were not granted to us).