.UPDATE() IN BUSINESS RULES

In short: Never do this.

In *long*: *Almost* never do this.

While there is **no** reason to use `current.update()` in a Business Rule (**or in any code that's called from a Business Rule**), there are some occasions when it's appropriate to use the `.update()` GlideRecord API method. Just not on `current`. Virtually every such exception applies to **after** Business Rules (Business Rules configured to run **after** the database operation that triggered it has finished).

If the record which triggered the BR needs to be updated in some way, that update should happen in a **before** Business Rule (which runs on the application server *before* the updated record is sent to the database). "Before" BRs effectively intercept the data on its way to the database server, running their code at that point, and *then* sending the resultant record (including any modifications) to the database (unless something like `current.setAbortAction(true)` is used). Because of this fact, using current.update() in a *before* Business Rule is neither necessary, nor wise – the record you've updated is *already on its way to the database*.

If you need to update a **peripheral** record (not the record that's in the `current` object in the Business Rule, but perhaps a referenced or related record), then it is reasonable to do so using `.update()`; **however**, updating of peripheral records should be done in an **after** Business Rule; not **before**. In fact, if the records that need updating are not directly referenced on the current record (or otherwise don't require the change to take place *immediately*, those changes should be made in an **async** Business Rule, so that the user who submitted the form that triggered the Business Rule doesn't have to wait for the completion of that peripheral update!

The reason it's important to not use `current.update()` in a Business Rule, is that to do so would trigger a **second** update of the record; one for the transaction that triggered the BR in the first place, and a second one for the `current.update()` line in the BR's code. This could even lead to an **infinite loop**! The second update could trigger the Business Rule to run again, which would then trigger another update, which triggers the BR again, and so on, ad infinitum. Talk about *performance-hell*!