

Item 115

git_comments:

git_commits:

1. **summary:** CAMEL-15176: Optimize component to do as much in init phase vs start phase.
message: CAMEL-15176: Optimize component to do as much in init phase vs start phase.

github_issues:

github_issues_comments:

github_pulls:

github_pulls_comments:

github_pulls_reviews:

jira_issues:

1. **summary:** Component - Use doInit for wiring instead of doStart (part 2)
description:

jira_issues_comments:

1. There is a file that tracks the status at <https://github.com/apache/camel/blob/master/components/components-init-work-in-progress.md>
2. Hey [~davsclaus], I have a question in regards to this, as far as I understood any client creation operations should be moved to `{{doInit}}` instead of `{{doStart}}` only for static stuff as long that it does not create threads, runtime depending operating ... etc, isn't? What if for example the client would create its own threads to interact with the service, for example [EventHub client]<https://github.com/apache/camel/blob/4ea9e6c357371682b855d2d79655b41120331b7a/components/camel-azure-eventhubs/src/main/java/org/apache/camel/component/azure/eventhubs/EventHubsConsumer.java#L47>, I'd assume in this case we just leave it under `{{doStart}}`? Or at least create the client in the `{{doInit}}` and start it under `{{doStart}}` ?
3. **body:** Yes exactly. init is most useable for dependency injection, loading resources from classpath, creating basic objects etc. And yes if a client as you talk about have a start method that creates threads, then that should be in doStart. But creating the client via new XXX may sometimes be possible in init, unless it starts threads etc in its constructor. So a good thing is to look inside the source of the client what it does to be sure.
label: code-design
4. Thanks for the tip, will keep it in mind (y)