Anforderungsbeschreibung:

Als zentrale Nachrichtenzentralle in BIKS wird ein Blackboard verwendet. Das Blackboard wird in Form eines Actors im SFA gehostet.

Auf dem Blackboard werden beispielsweise Befehle wie Alarm, UserInput oder Shutdown hinzugefügt. Der Controller verarbeitet die Befehle und gibt sie an die KnowledgeSourcen weiter.

Dort wird versucht mit allen vorhandenen KPUs das Problem zu lösen.

Kann ein Problem von einem Blackboard nicht gelöst werden, muss dieses eventuell auf ein anderes Blackboard gesendet werden.

Anbei in der Tabelle seinen alle Befehlsobjekte und Knowledge Sources exemplarisch aufgelistet:

|  |  |  |
| --- | --- | --- |
| Bezeichnung | Befehl/KU | Beschreibung |
| AlarmObject | Blackboard Befehl | Es wurde ein Alarm ausgelöst. Z.B. von Jarvis |
| AskUserInput | Blackboard Befehl | Es soll ein Benutzerdialog aufgehen. |
| ShutdownObject | Blackboard Befehl | Anlage soll abgeschaltet werden. |
| Alarm | Knowledge Source | Alarm Einheit bearbeitet Alarm Objekt und weiß welche Agenten im Alarmfall zu benachrichtigen sind |
| UserInput | Knowledge Source | UserInput öffnet einen Benutzerdialog und hält Rücksprache mit dem Benutzer. |
| SupportedCommand | Knowledge Source | Prüft ob es sich bei dem an das Blackboard übergebene Kommando um ein unterstütztes Kommando handelt. |
| ShutdownFacility | Knowledge Source | Kommando verarbeitet das ShutdownObject Z.B. Stillstand der Maschine, Logging auf Konsole |
| NotProcessed | Knowledge Source | Überprüft nochmals ob alle Befehle abgearbeitet worden sind. |

Use Case: „Alarm wird ausgelöst“

1. JARVIS sendet ein gültiges AlarmObject an das Blackboard.
2. Der Controller durchläuft alle seine Knowledge Units.
3. Der Befehl ist im Zustand Queued wird von der SupportedCommand KU verarbeitet und auf Supported gesetzt.
4. Im nächsten Durchlauf wird der auf Supported gesetzte Alarm Befehl von der Alarm KU behandelt. Hier würden bei einem realen System Befehle zur Alarmierung abgesetzt.
5. In diesem System wird ein AskUserInput Befehl an das Blackboard gesendet.
6. Der AskUserInput Befehl wird von der SupportedCommand KU bearbeitet.
7. Im weiteren Schritt wird der AskUserInput Befehl von der UserInput KU bearbeitet und eine Benutzerinteraktion mit einem Dialog erzeugt.
8. Wird der Dialog positiv quittiert, wird ein Shutdown Objekt auf das Blackboard gelegt.
9. Dieses wird dann analog zu Alarm und UserInput von SupportedCommand und seiner zuständigen ShutdownFacility KU verarbeitet.
10. Zuletzt wird die NotProcessed KU durchlaufen und nicht vollständig verarbeitete Befehlsobjekte z.B. geloggt.

ActorFabric:

Der ActorFabric wird nach der Fabriksmethode den jeweils erforderlichen Actor für die jeweils auszuführende Activity instanziieren. Als Eingangsparameter wird eine Referenz auf einen Sender und der jeweilige Befehlstyp übergeben. Intern werden beide Eingangsparameter gespeichert. Der Zeiger auf den Actor wird zurückgegeben.

Das Blackboard wird mit dem Actor Fabric eng zusammenarbeiten.