Anforderungsdefinition

RD 18-10/01 – Demo Fertigungszelle – Step I



Neutorstraße 13  
5020 Salzburg

AUSTRIA

Tel: +43 (662) 276198-11

Fax: +43 (662) 276198-98

Mail: [office@breanos.com](mailto:office@breanos.com)

|  |  |
| --- | --- |
| Datei | RD 18-1001 Demo Zellenfertigung - Step I.docx |
| Datum | 30.10.2018 |

Inhaltsverzeichnis

[1 Änderungsverzeichnis 3](#_Toc526852816)

[2 Anforderungsbeschreibung 4](#_Toc526852817)

[2.1 Referenzen 4](#_Toc526852818)

[2.2 Einführung 4](#_Toc526852819)

[2.2.1 Inkludierter Umfang 4](#_Toc526852820)

[2.2.1.1 Simulation einer Zelle 5](#_Toc526852821)

[2.2.1.2 Verarbeitende KPUs 5](#_Toc526852822)

[2.2.1.3 Anzeige 5](#_Toc526852823)

[2.2.2 Exkludierter Umfang 6](#_Toc526852824)

[2.3 Geplantes Vorgehen 6](#_Toc526852825)

# Änderungsverzeichnis

|  |  |  |  |
| --- | --- | --- | --- |
| Datum | Version | Verfasser | Beschreibung |
| 27.09.2018 | 1.0 | Dominik Hutterer | Erstversion mit rudimentären Basisfunktionalitäten. |
| 09.10.2018 | 1.1 | Dominik Hutterer | Überarbeitung Dokumentstruktur |
| 24.10.2018 | 2.0 | Dominik Hutterer | Vollkommene Überarbeitung, hin auf einen möglichen Test Case mit der FH Salzburg. |

# Anforderungsbeschreibung

## Referenzen

Das Daipan Framework in der aktuellen Version.

Konzepte, Datenstrukturen, Abläufe, etc. aus den vergangenen TTS/PTS Projekten.

## Einführung

Zu Demonstrationszwecken und für interne Tests wird eine Demoapplikation für das Framework erstellt, deren Aufgabenstellung nahe an bestehenden Projekten liegt und somit ein möglichst reales Anforderungsprofil wiederspiegelt.

Das Projekt umfasst sowohl den Simulationsteil, wie auch den eigentlichen verarbeitenden Teil. Beide Teile werden im aktuell definierten KPU/CWF Design realisiert und in eine zuvor vollständig kalibrierte Testumgebung integriert. Die Entwicklung erfolgt stufenweise in überschaubaren Einzelschritten, so soll aus einer anfänglich rudimentären Anwendung ein zunehmend komplexeres System entstehen. Das Projekt wird somit in mehrere Phasen aufgeteilt und für jede Phase der Erweiterungsumfang der vorhergehenden Phase festgelegt. Dies geschieht in sogenannten Change-Request-Dokumenten, wobei für jede Phase ein neuer Change-Request erstellt wird.

Um eine möglichst flexible und zukunftsorientierte Aufgabenstellung zu gewährleisten, ist eine Zellenfertigung bzw. eine Zelle einer Zellenfertigung als Ausgangspunkt gewählt worden.

### Inkludierter Umfang

Der angedachte Show Case der FH Salzburg besteht aus **zwei Zellen** und **einer Transporteinheit**, die folgende Struktur aufweisen:

Zelle 1:

* 3D-Drucker 🡪 OP10
* Kooperativer Roboter 🡪 OP20

Transport:

* Autonom fahrendes Transportsystem

Zelle 2:

* Roboter 🡪 OP30

Die Verarbeitung in den Maschinen erfolgt grundsätzlich parallel, jede Maschine kann jedoch immer nur eine Aktion nach der anderen ausführen.

Zur Vereinfachung erfolgt der Datenaustausch zwischen Simulation und dem BREANOS-System direkt über das Blackboard ohne Zwischenschicht (TCP/OPC/…).

#### Schnittstellendefinition

Für eine einheitliche Umsetzung sowohl für den Show Case, wie auch für die Simulation, müssen die Schnittstellen noch genauer definiert werden.

Dies umfasst folgendes:

* Abklärung der verwendeten Hardwarekomponenten und deren Hersteller (KUKA, Universal Robots, …)
* Abklärung der möglichen Schnittstellentechnologien (bevorzugt wird OPC UA)
* Abklärung der Aufgabenbereiche und der zu übermittelnden Daten.
* Festlegen von Protokollen/Technologien und Datenstrukturen

#### Verarbeitende KPUs

Für jede der oben genannten Maschinen muss eine entsprechende verarbeitende KPU geschaffen werden, welche sowohl den Maschinenstatus wiederspiegelt – aktuell den aktuellen Status (Aktiv, Warten, Offline …), die Werkstücknummer, den Namen und einen Stückzähler, wie auch die möglichen Aktivitäten der Maschine implementiert. Ein übergeordneter Workflow übernimmt die Steuerung

Zusätzlich benötigt es eine TTS KPU in welche die Maschinen KPUs bestenfalls integriert werden. Gut wäre auch eine zusätzliche Zusammenfassung in einer Zelle, um später zusätzliche Zellen einfach einzubinden.

Die TTS KPU empfängt vom BB ein StatusUpdate und legt dieses in der Datenbank ab. Die weiteren Informationen (Werkstücknummer) werden an die entsprechende KPU weitergeleitet 🡪 Umschlüsselung Location auf eindeutigen Namen.

Empfängt eine Maschinen eine neue Werkstücknummer, wird diese als aktuelles Werkstück übernommen und der Stückzähler inkrementiert.

Die Daten der Maschine werden persistiert, um beim Neustart verfügbar zu sein.

Eine Funktionalität zum Zurücksetzen des Werkstückzählers soll implementiert werden.

#### Simulation

Via CWF soll eine Produktion auf zwei verbundenen Zellen simuliert werden.

Für die erste Umsetzung sollte die Struktur möglichst simpel gehalten werden.

Folgendes wäre denkbar:

* Eine Aktivität zur Generierung eines neuen Werkstücks:
  + Eine neue Werkstücknummer wird generiert.
* Drei gleiche Aktivitäten, die jeweils eine Bearbeitung in einer Maschine darstellen, werden gestartet. Diese müssen mit den benötigten Parametern befüllt werden, bzw. darauf zugreifen können:
  + - Werkstücknummer
    - Operation
    - Location (Line/Zone/Place/Fixture)
    - Taktzeit in sek.
  + Wurde keine Werkstücknummer vergeben, beendet sich die Aktivität sofort.
  + Ein Part Status DTO wird mit diesen Informationen auf die entsprechende Queue gelegt (UpdateEvent = 1 – drop; Status = 3).
  + Der verarbeitende Thread wartet Taktzeit + X sek. (Random von -1s bis +8s)
  + Ein Part Status DTO wird mit diesen Informationen auf die entsprechende Queue gelegt (UpdateEvent = 1 – drop, Status = 3).
* Warten bis alle drei Aktivitäten abgeschlossen sind.
* Weiter Takten der Werkstücke Neu 🡪 M1 🡪 M2 🡪 M3 🡪 EOL
* Ein Part Status DTO für das EOL-Teil wird mit den vorhandenen Informationen auf die entsprechende Queue gelegt (UpdateEvent = 9 – eol; Status = 3, Location = letzte Location des Werkstücks).
* Wiederholung des Prozesses bis X Stück EOL gemeldet wurden.
* Die zuletzt generierte Werkstücknummer muss persistiert werden um nach dem Neustart keine doppelten Einträge zu bekommen.

#### Anzeige

Entsprechend der KPU Struktur muss ein Menü mit entsprechenden Subelementen abgebildet werden TTS 🡪 (Line 1) öffnet eine Linienübersicht – 1 Zelle mit 3 Maschinen & 1 Gantry. Es kann sowohl die Zelle, wie auch eine der Maschinen ausgewählt und angezeigt werden.

Die Zelle zeigt wiederum die beinhaltenden Elemente an, die Maschinen zeigen ihre Parameter an und bieten die Möglichkeit den jeweiligen Stückzähler zurückzusetzen.

**Option für Fleißige**: Bei Auswahl eines Werkstücks wird dessen Bearbeitungsgeschichte angezeigt.

### Exkludierter Umfang

Alle nicht in Punkt 2.2.1 explizit angeführten Anforderungen werden ausgeschlossen.

## Geplantes Vorgehen

Idealerweise umfasste jede Phase einen Sprint und besteht aus:

* Planung – Prosaideen und TFS User Stories
* Sprintstart
* Modellierung
* Implementierung
* Deployment & Testing

Am Ende des Sprints werden die einzelnen Elemente der Phase abgenommen.