

SimpleMilestonesHandler

Berechnet die Bearbeitungszeit von Aktivitäten neu, um gegebene Meilensteine zu erreichen. Meilensteine dürfen keine Scriptnodes sein und müssen eine Duration haben.

Der Actionhandler geht je Meilenstein folgendermaßen vor (es wird mit dem ersten begonnen):

1. Der Actionhandler geht alle Pfade bis zum jeweiligen Meilenstein durch und addiert dabei die Duration der jeweiligen Aktivitäten
2. Die Pfade werden ihrer Gesamtduration nach absteigend sortiert
 - 2.1. Falls soeben eine Aufgabe erledigt wurde, wird die Duration der gerade erledigten Aktivität auf die tatsächlich benötigte Bearbeitungszeit gesetzt.
 - 2.2 Es wird mit dem längsten verbleibenden Pfad begonnen.
 - 2.3 Dabei wird die Gesamtduration erneut berechnet (ggf. mit bereits gekürzten Durations) und überprüft, ob das Datum des Meilensteins eingehalten werden kann.
 - 2.3.1 Wenn das Datum eingehalten wird, wird bei 1. mit dem nächsten Meilenstein fortgefahren.
 - 2.3.2 Kann der übergebene Termin nicht mit den aktuellen Durations erreicht werden, wird errechnet, wieviel Prozent der Duration allen Aktivitäten des Pfades abgezogen werden muss, um den Meilenstein zu erreichen. Der Prozentsatz wird nun von den Durations aller noch nicht beendeter Aktivitäten des Pfades abgezogen. Anschließend wird der kritische Pfad neu berechnet.

Achtung! Soll die Berechnung nach jeder Aktivität stattfinden empfiehlt es sich den [AddActionHandlerToNodes](#) zu verwenden. In iGrafx muss die Funktion „Kritischen Pfad berechnen“ aktiviert sein

Klasse

```
com.dooris.bpm.actionhandler.SimpleMilestonesHandler
```

Event Type

node-leave WICHTIG!

Action Name

beliebig

Mandatory Fields

leer

Parameter

path

Pfad zu txt-Datei, in welcher die Meilensteine hinterlegt sind.
In der txt-Datei sind alle möglichen Pfade bis zu den Meilensteinen hinterlegt.
Beispiel

```
path=/jboss_tim/server/default/deploy/loom.ear/loom-portal.war/custom/super/milestones.txt;
```

Erklärung (siehe [Beispiel TXT Datei](#))

dates

IDs der Felder in welche die Daten geschrieben werden.
Gibt es mehre Meilensteine, werden diese kommagetrennt übergeben.

Beispiel

```
dates=${milestone_1},${milestone_2},${milestone_3};
```

Erklärung (siehe [Beispiel](#))

dateformat

Das Format, in welchem das Datum der Meilensteine formatiert ist. Hier wird SimpleDateFormat verwendet.

```
dateformat=dd.MM.yyyy HH:mm;
```

Erklärung (siehe [Beispiel](#)). Mögliche Optionen sind:

Beispiel TXT Datei

Erklärung: Mit # können Zeilen auskommentiert werden. Die Aktivitäten werden kommagetrennt übergeben. Pro Pfad wird eine neue Zeile verwendet.

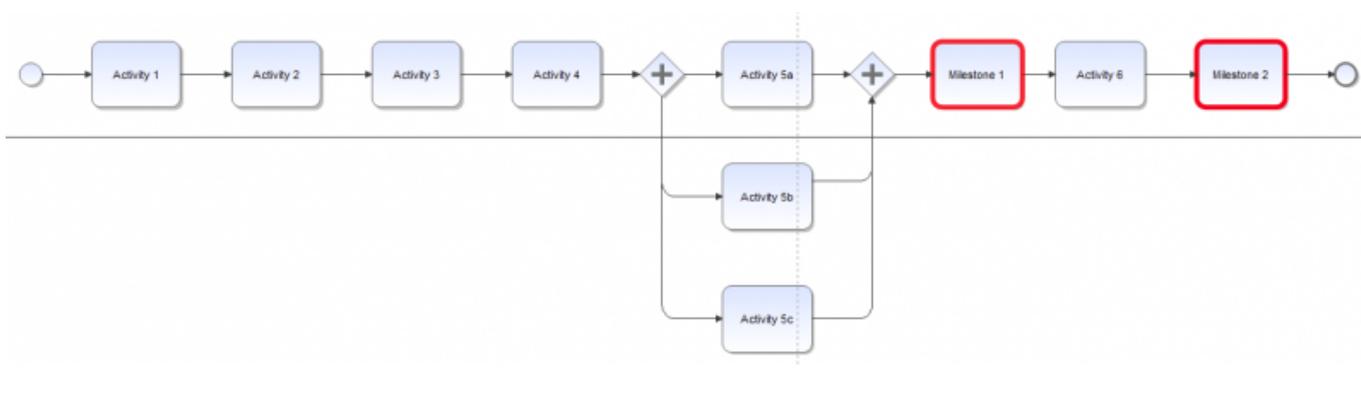
Meilensteine werden mit \$ maskiert. Der erste Parameter gibt an, welches Datum für den Meilenstein greift. (0-n)

```

# Use # to mark a comment row
#Possible paths to Milestone 0 - Milestone 1
0,Activity 1,Activity 2,Activity 3,Activity 4,Activity 5a,$Milestone 1
0,Activity 1,Activity 2,Activity 3,Activity 4,Activity 5b,$Milestone 1
0,Activity 1,Activity 2,Activity 3,Activity 4,Activity 5c,$Milestone 1
#Possible paths to Milestone 1 - Milestone 2
1,Activity 1,Activity 2,Activity 3,Activity 4,Activity 5a,Milestone 1,Activity 6,$Milestone 2
1,Activity 1,Activity 2,Activity 3,Activity 4,Activity 5b,Milestone 1,Activity 6,$Milestone 2
1,Activity 1,Activity 2,Activity 3,Activity 4,Activity 5c,Milestone 1,Activity 6,$Milestone 2

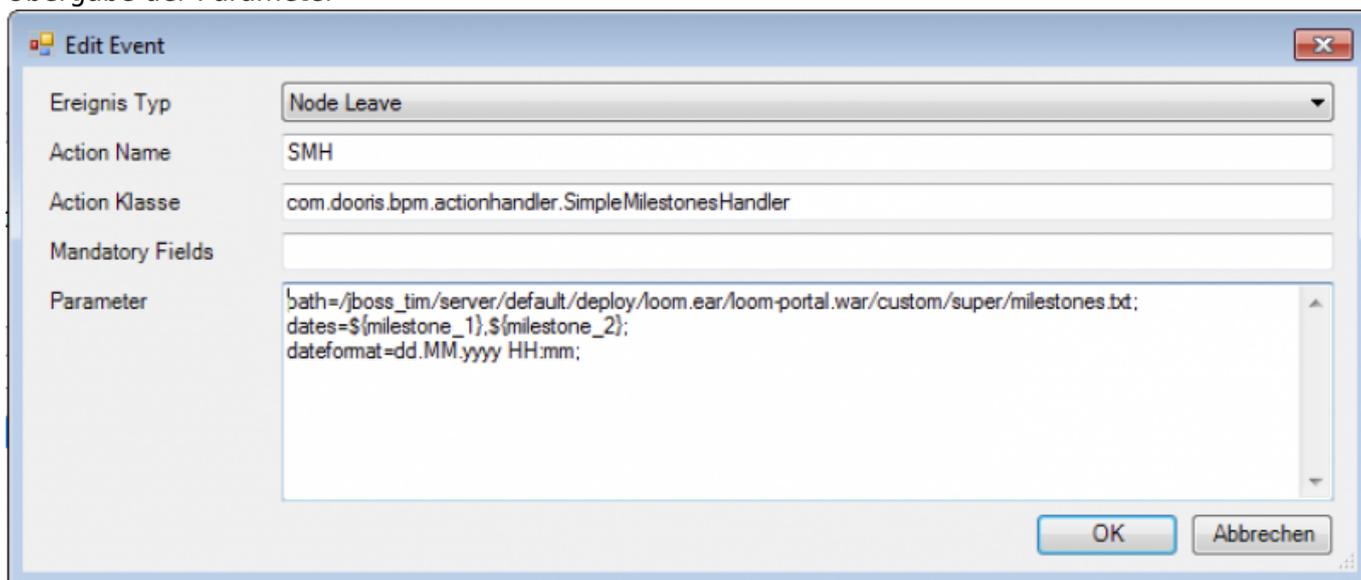
```

Das Beispiel trifft auf folgenden Prozess zu.



Beispiel

Übergabe der Parameter



Abhängigkeiten

Verfügbar ab [Version 4.3](#)

From: <https://wiki.tim-solutions.de/> - [TIM Wiki](#) / [NEW TIM 6 Documentation](#)

Permanent link: <https://wiki.tim-solutions.de/doku.php?id=software:tim:actionhandler:simplemilestoneshandler&rev=1455274627>

Last update: **2021/07/01 09:57**

