

Kurze Formalisierung Java-ALED:

Episode := Block {Block}

Satz: Der Ablauf erfolgt chronologisch von oben nach unten innerhalb einer Episode!

Block := '(' [Auftrittshäufigkeit] 'EOL' Block ')' 'EOL'
oder Block Abfolgeoperator [Auftrittshäufigkeit] 'EOL' Block
oder Anweisung 'EOL' {Anweisung 'EOL'}

Abfolgeoperator := 'SEQ'
oder 'ALTER'
oder 'COMB'

Anweisung := Auftrittshäufigkeit '{ Ereignismenge }' '/' {Aktionsanweisung}

Auftrittshäufigkeit := " Mindesthäufigkeit ',' Höchsthäufigkeit "

Mindesthäufigkeit := '0'
oder '1'
oder '2'
oder ...
Höchsthäufigkeit := '1'
oder '2'
oder '3'
oder ...
oder '*'

Satz: (Höchsthäufigkeit >= Mindesthäufigkeit) oder (Höchsthäufigkeit == *)

Ereignismenge := Ereignistyp
oder Ereignismenge Mengenoperator Ereignismenge
oder '{ Ereignismenge }' [Bedingung]

Mengenoperator := '+'
oder '\'
oder '~'

Ereignistyp := Ereignisname [Bedingung]

Ereignisname := '#'

oder '<' mögliches Namensattribut eines Ereignisses '>'

Bedingung := '[' Boole'scher Ausdruck ']

Satz: <abc> == #[event.name == abc]

Satz: <abc>[true] == <abc>

Aktionsanweisung := Anweisung in Form der zugrunde liegenden Sprache

Legende:

rot : Zeichen die lediglich zur Verdeutlichung der Formalisierung benutzt werden und nichts mit der eigentlichen Schreibweise zutun haben.

blau : Formalisierungsbeschreibung

grün : Erläuterungen (ggf. Verweis auf weitere Formalisierungen z.B. Java)

schwarz : endgültige Werte oder Zeichen

unterstrichen : Sätze

Bsp: (Block aus einer Anweisung)

'1,5' {#[true] \ <run-debug>[event.file == „xyz“]} / String s = „habe-was“;

Mindesthäufigkeit: 1

Höchsthäufigkeit: 5

Ereignismenge: #[true] \ <run-debug>[event.file == „xyz“]

=> #[true] ist Ereignismenge, \ ist Mengenoperator, <run-debug>[event.file == „xyz“] ist Ereignismenge.

Wobei # und <run-debug> Ereignisnamen und true sowie event.file == „xyz“ Bedingungen sind.

String s = „habe-was“; ist Aktionsanweisung