

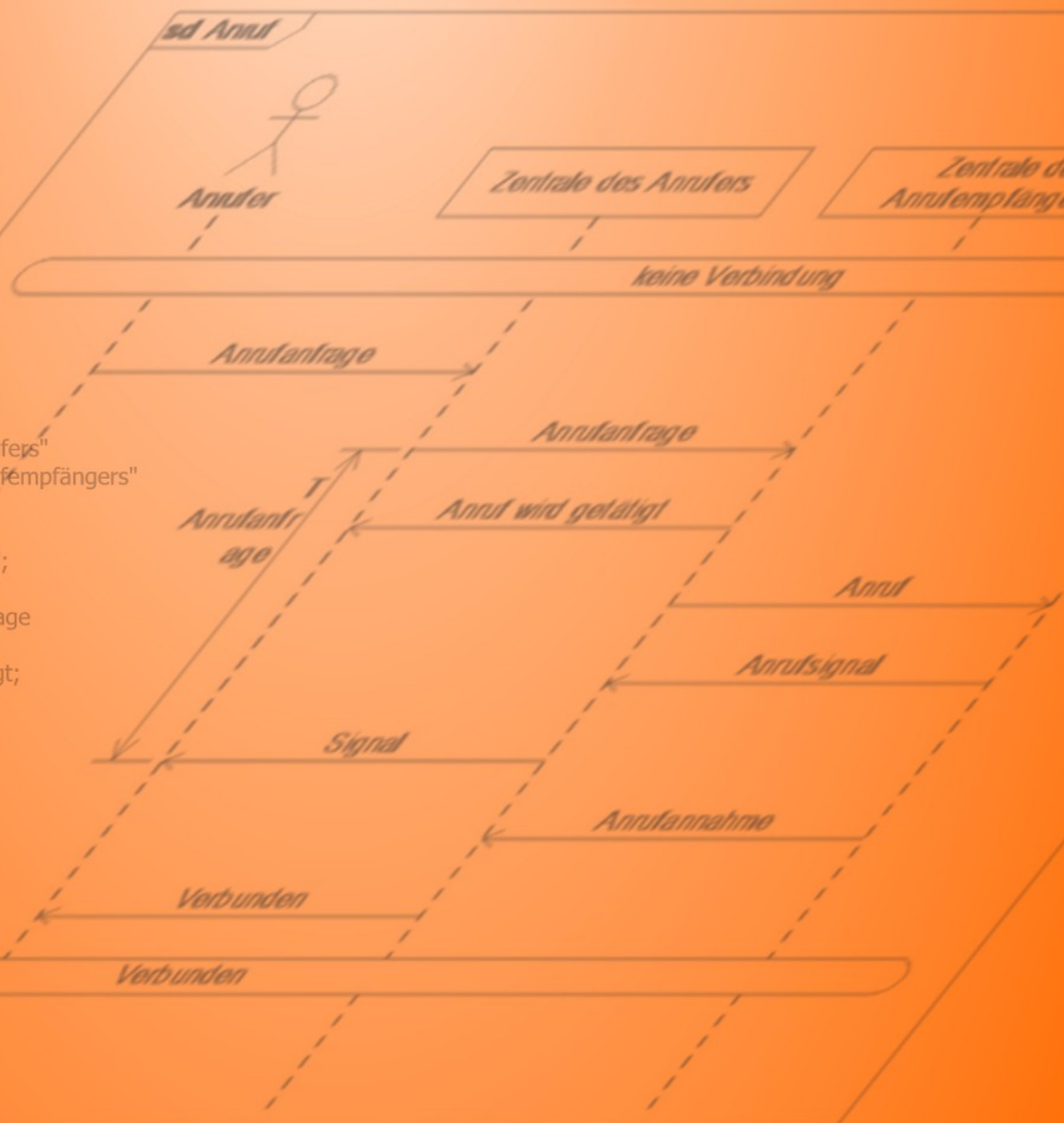
# Sequenzdiagramm Generator

## Befehlssatz

PrintAuthor: yes  
 PrintCompany: yes  
 PrintVersion: yes  
 PrintDate: yes  
 PrintCreationDate: yes  
 PrintFileName: yes  
 Font: "Arial", "10", "Regular"  
 DiagramName: Anruf  
 DiagramStyle: uml  
 PageMargins: 5, 5, 5, 5  
 PrintFootLine: yes  
 Author: Franz Mustermann  
 Company: ITESYS GmbH  
 Version: 1.0  
 Date: 20.10.2006  
 PageSize: 660, auto

actor: CLU, "Anrufer"  
 process: ON, "Zentrale des Anrufers"  
 process: DN, "Zentrale des Anrufempfängers"  
 actor: CDU, "Anrufempfänger"

stateoverall: "keine Verbindung";  
 msg: CLU, ON, "Anrufanfrage";  
 timerbegin: t1, ON, T,Anrufanfrage  
 msg: ON, DN, Anrufanfrage;  
 msg: DN, ON, Anruf wird getätigt;  
 msg: DN, CDU, Anruf;  
 msg: CDU, DN, Anrufsignal;  
 msg: DN, ON, Signal  
 timerend: t1;  
 msg: CDU, DN, Anrufannahme;  
 msg: DN, ON, Verbunden;  
 stateoverall: Verbunden;

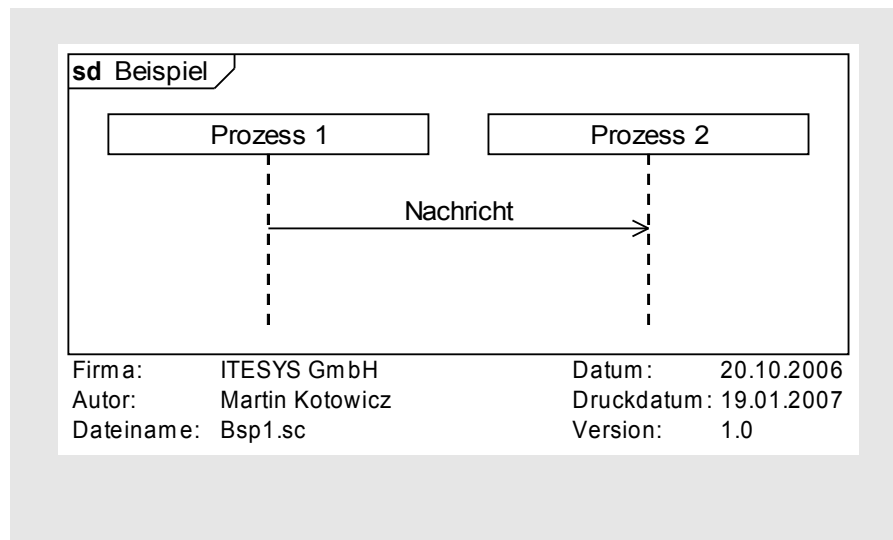


<b>1</b>	<b>FORMATIERUNG</b>	<b>4</b>
1.1	SEITENFORMATIERUNG	4
1.1.1	PAGESIZE	4
1.1.2	PAGEMARGINS	7
1.2	DIAGRAMMFORMATIERUNG	8
1.2.1	DIAGRAMNAME	8
1.2.2	DIAGRAMSTYLE	8
1.2.3	FONT	9
1.2.4	LEFT	10
1.2.5	RIGHT	11
1.2.6	LINEOFFSET	11
1.2.7	NEXTPAGE	13
1.3	FUSSZEILE	13
1.3.1	PRINTFOOTLINE	13
1.3.2	AUTHOR	14
1.3.3	PRINTAUTHOR	14
1.3.4	COMPANY	15
1.3.5	PRINTCOMPANY	15
1.3.6	PRINTFILENAME	16
1.3.7	DATE	16
1.3.8	PRINTDATE	17
1.3.9	PRINTCREATIONDATE	17
1.3.10	VERSION	18
1.3.11	PRINTVERSION	18
1.4	FARBENDEFINITION	18
1.4.1	BACKCOLOR	19
1.4.2	FILLCOLOR	19
1.4.3	TEXTCOLOR	20
<b>2</b>	<b>DIAGRAMMELEMENTE</b>	<b>21</b>
2.1	LEBENSLINIE	21
2.1.1	ACTOR	21
2.1.2	PROCESS	22
2.1.3	DUMMYPROCESS	23
2.1.4	CREATE	24
2.1.5	STOP	24
2.1.6	REGIONBEGIN	25
2.1.7	REGIONEND	26
2.2	NACHRICHTEN	26
2.2.1	MSG	27
2.2.2	MSGBEGIN	28
2.2.3	MSGEND	30
2.2.4	FOUND	30
2.2.5	LOST	31
2.3	ZUSTANDSINVARIANTEN	32
2.3.1	STATE	32
2.3.2	STATEOVERALL	33
2.4	TÄTIGKEITEN	34
2.4.1	TASK	34
2.5	KOMMENTARE	35
2.5.1	COMMENT	35
2.5.2	COMMENTOVERALL	36
2.5.3	LINECOMMENT	37
2.5.4	MARK	38
2.6	TIMER	39
2.6.1	TIMERBEGIN	39
2.6.2	TIMEREND	41
2.6.3	SETTIMER	41
2.6.4	STOPTIMER	42
2.6.5	TIMEOUT	43

2.7 ZEITMESSUNGEN.....	44
2.7.1 MEASUREBEGIN.....	44
2.7.2 MEASUREEND.....	46
2.7.3 MEASURESTART.....	46
2.7.4 MEASURESTOP.....	48
2.8 KOMBINIERT E FRAGMENTE.....	49
2.8.1 FRAGMENTBEGIN.....	49
2.8.2 FRAGMENTEND.....	50
2.8.3 FRAGMENTSEPARATOR.....	50
2.8.4 FRAGMENTTEXT.....	51
2.9 REFERENZEN.....	51
2.9.1 REF.....	51
<b>3 STEUERZEICHEN.....</b>	<b>52</b>
3.1 TEXTFORMATIERUNG.....	52
3.1.1 \n – Zeilenumbruch.....	52
3.1.2 \“ – Anführungszeichen.....	53
3.2 DIAGRAMMFORMATIERUNG.....	53
3.2.1 ; – Diagrammzeilenumbruch.....	53
3.2.2 { } – kein Diagrammzeilenumbruch.....	54
3.2.3 ;; – manueller Seitenumbruch.....	55
<b>ANHANG A.....</b>	<b>56</b>
FARBENNAMEN.....	56
<b>4 ANHANG B.....</b>	<b>58</b>
INDEX.....	58

## 1 Formatierung

Die Formatierung des Diagramms beinhaltet das Festlegen des Diagrammnamens, Fußzeile, des Darstellungsstandards, der verwendeten Schriftart und der Diagrammgröße. Dieses Kapitel listet die Befehle auf, die für die Formatierung des Diagramms verantwortlich sind.



**DiagramName:** Beispiel  
**DiagramStyle:** uml  
**Font:** "Arial", "10", "Regular"  
**PrintAuthor:** yes  
**PrintCompany:** yes  
**PrintVersion:** yes  
**PrintDate:** yes  
**PrintCreationDate:** yes  
**PrintFileName:** yes  
**PageMargins:** 5, 5, 5, 5  
**PrintFootLine:** yes  
**Author:** Martin Kotowicz  
**Company:** ITESYS GmbH  
**Version:** 1.0  
**Date:** 20.10.2006  
**PageSize:** 400, auto

**process:** p1, Prozess 1  
**process:** p2, Prozess 2  
**msg:** p1, p2, Nachricht;

### 1.1 Seitenformatierung

Die Seitenformatierung erlaubt es die Eigenschaften der Ausgabeseite zu verändern. Mit diesen Befehlen lassen sich die Seitengröße und die Seitenränder des Diagramms definieren.

#### 1.1.1 PAGESIZE

Über den Befehl **pagesize** kann die Größe des Diagramms festgelegt werden. Der Befehl stellt zwei unterschiedliche Parametersätze zur Verfügung, die je nach Bedarf des Anwenders verwendet werden können. Eine direkte Eingabe der Seitengröße von A0 bis A5 kann durch den Parameter *size* erfolgen, die zugehörige Orientierung kann durch den Parameter *orientation* angepasst werden. Der zweite Parametersatz ermöglicht eine eigenständige Definition der Diagrammbreite und Diagrammhöhe durch die Parameter *width* und *height*. Eine Festlegung der Größeneinheit der beiden Parameter ist durch den optionalen Parameter *unit* möglich.

#### ➤ Syntax

**pagesize:** size, orientation

oder

**pagesize:** width, height, unit

➤ **Parameter**

- size

Wert	Bedeutung
A0	Das Diagramm hat die Größe einer DIN A0 Seite (118,9cm x 84,1cm)
A1	Das Diagramm hat die Größe einer DIN A1 Seite (84,1cm x 59,4cm)
A2	Das Diagramm hat die Größe einer DIN A2 Seite (59,4cm x 42cm)
A3	Das Diagramm hat die Größe einer DIN A3 Seite (42cm x 29,7cm)
A4	Das Diagramm hat die Größe einer DIN A4 Seite (29,7cm x 21cm)
A5	Das Diagramm hat die Größe einer DIN A5 Seite (21cm x 14,8cm)

- orientation (optional)

Wert	Bedeutung
h	Querformat
v	Hochformat

Der Standardwert für den Parameter ist "h".

- width

Dieser Parameter legt die Breite des Diagramms fest. Die Größeneinheit ist vom Parameter `unit` abhängig. Bei der Eingabe von Dezimalzahlen muss beachtet werden, dass diese in Anführungszeichen angegeben werden. Der Wert "auto" passt die Breite automatisch an die Anzahl der definierten Prozesse an.

- height

Dieser Parameter legt die Höhe des Diagramms fest. Die Größeneinheit ist vom Parameter `unit` abhängig. Bei der Eingabe von Dezimalzahlen muss beachtet werden, dass diese in Anführungszeichen angegeben werden. Der Wert "auto" passt die Höhe des Diagramms automatisch an dessen Länge an.

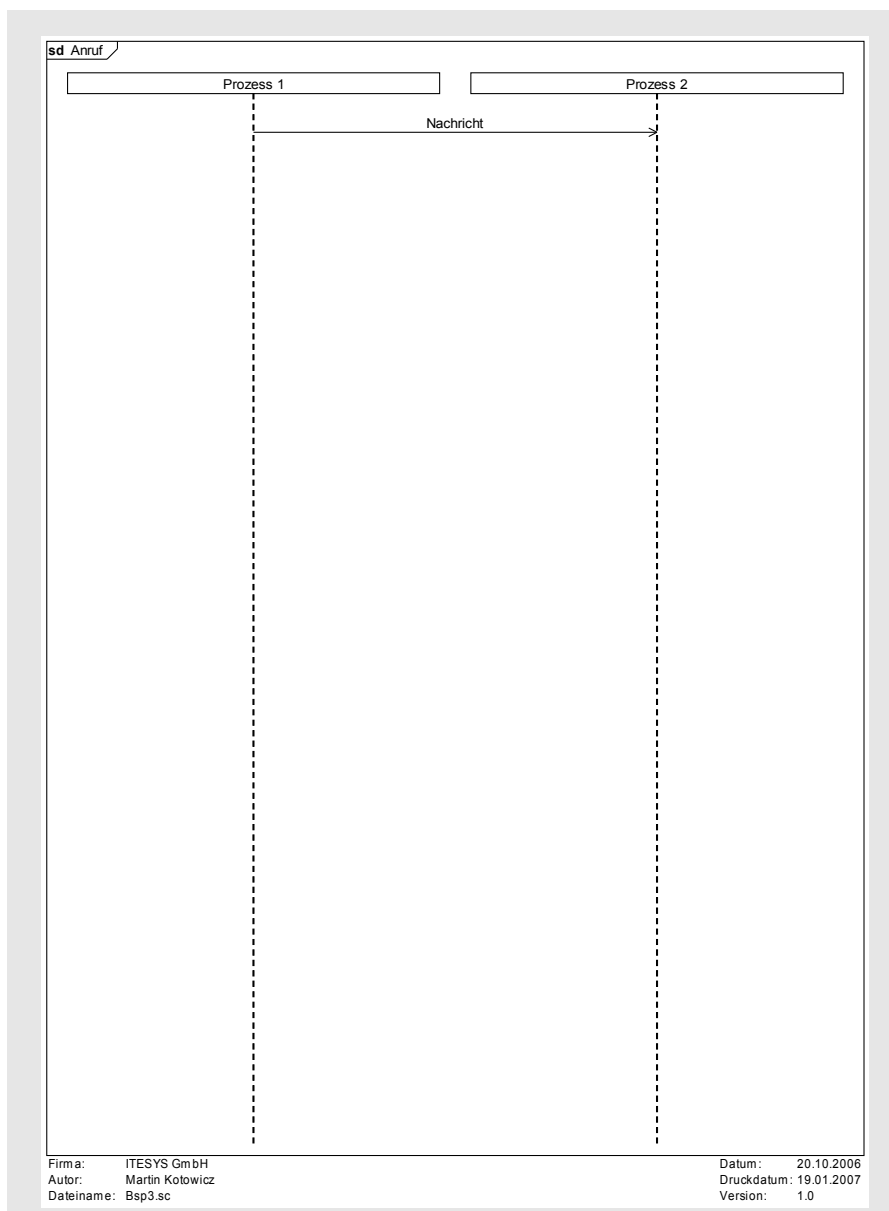
- unit (optional)

Dieser Parameter definiert die Größeneinheit der unter `width` und `height` angegebenen Werte. Für den wert "auto" ist dieser Parameter ohne Bedeutung.

Wert	Bedeutung
pixel	Bildpunkte (Pixel)
mm	Millimeter
cm	Zentimeter
inch	Zoll

Der Standardwert für den Parameter ist "pixel".

➤ **Beispiel**



**DiagramName:** Anruf  
**DiagramStyle:** uml  
**Font:** "Arial", "10", "Regular"  
**PrintAuthor:** yes  
**PrintCompany:** yes  
**PrintVersion:** yes  
**PrintDate:** yes  
**PrintCreationDate:** yes  
**PrintFileName:** yes  
**PageMargins:** 5, 5, 5, 5  
**PrintFootLine:** yes  
**Author:** Martin Kotowicz  
**Company:** ITESYS GmbH  
**Version:** 1.0  
**Date:** 20.10.2006

**PageSize:** A4, v

**process:** p1, Prozess 1  
**process:** p2, Prozess 2  
**msg:** p1, p2, Nachricht

## 1.1.2 PAGEMARGINS

Über den Befehl **pagemargins** kann die Größe der Diagrammränder definiert werden. Die Diagrammränder werden von der mit dem Befehl **pagesize** definierten Gesamtgröße des Diagramms abgezogen und verkleinern dadurch die graphische Diagrammausgabe. Die Größe der Diagrammränder kann über die Parameter **left**, **top**, **right**, **bottom** und **unit** definiert werden. Der Parameter **unit** ist optional. In diesem Fall wird der Standardwert "pixel" verwendet.

### ➤ Syntax

**pagemargins**: left, top, right, bottom, unit

### ➤ Parameter

- left

Dieser Parameter definiert die Breite des linken Randes. Die Größeneinheit ist vom Parameter **unit** abhängig. Bei der Eingabe von Dezimalzahlen muss beachtet werden, dass diese in Anführungszeichen angegeben werden. Der Standardwert für den Parameter ist "10".

- top

Dieser Parameter definiert die Höhe des oberen Randes. Die Größeneinheit ist vom Parameter **unit** abhängig. Bei der Eingabe von Dezimalzahlen muss beachtet werden, dass diese in Anführungszeichen angegeben werden. Der Standardwert für den Parameter ist "10".

- right

Dieser Parameter definiert die Breite des rechten Randes. Die Größeneinheit ist vom Parameter **unit** abhängig. Bei der Eingabe von Dezimalzahlen muss beachtet werden, dass diese in Anführungszeichen angegeben werden. Der Standardwert für den Parameter ist "10".

- bottom

Dieser Parameter definiert die Höhe des unteren Randes. Die Größeneinheit ist vom Parameter **unit** abhängig. Bei der Eingabe von Dezimalzahlen muss beachtet werden, dass diese in Anführungszeichen angegeben werden. Der Standardwert für den Parameter ist "10".

- unit (optional)

Dieser Parameter definiert die Größeneinheit für die unter **left**, **top**, **right** und **bottom** angegebenen Werte.

Wert	Bedeutung
pixel	Bildpunkte (Pixel)
mm	Millimeter
cm	Zentimeter
inch	Zoll

Der Standardwert für den Parameter ist "pixel".

## 1.2 Diagrammformatierung

Die Diagrammformatierung erlaubt es Einfluss auf die Darstellung des generierten Diagramms zu nehmen. Mit Hilfe der Befehle lässt sich unter anderem der Darstellungsstandard (SDL/UML) und der Name des Diagramms festlegen.

### 1.2.1 DIAGRAMNAME

Über den Befehl **diagramname** kann der Name des Diagramms definiert werden. Der Diagrammname wird durch den Parameter `name` definiert und erscheint in der linken oberen Ecke des Diagramms. Der Prefix "sd" oder "MSC" wird automatisch generiert.

#### ➤ Syntax

**diagramname:** name

#### ➤ Parameter

- name

Der Diagrammname. Der Wert des Parameters darf eine beliebige alphanumerische Zeichenfolge sein.

### 1.2.2 DIAGRAMSTYLE

Über den Befehl **diagramstyle** kann der Darstellungsstandard festgelegt werden. Der Darstellungsstandard wird durch den Parameter `style` definiert.

#### ➤ Syntax

**diagramstyle:** style



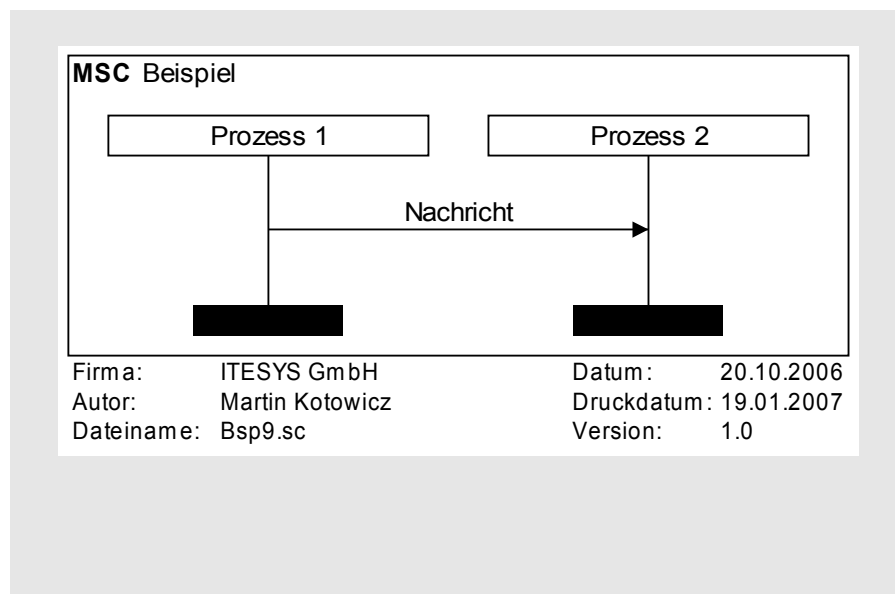
➤ **Parameter**

- style

Wert	Bedeutung
uml	Das Diagramm wird in UML2.0 generiert.
sdl	Das Diagramm wird in SDL generiert.

Der Standardwert für den Parameter ist "uml".

➤ **Beispiel**



**DiagramName:** Beispiel

**DiagramStyle:** sdl

**Font:** "Arial", "10", "Regular"

**PrintAuthor:** yes

**PrintCompany:** yes

**PrintVersion:** yes

**PrintDate:** yes

**PrintCreationDate:** yes

**PrintFileName:** yes

**PageMargins:** 5, 5, 5, 5

**PrintFootLine:** yes

**Author:** Martin Kotowicz

**Company:** ITESYS GmbH

**Version:** 1.0

**Date:** 20.10.2006

**PageSize:** 400, auto

**process:** p1, Prozess 1

**process:** p2, Prozess 2

**msg:** p1, p2, Nachricht;

### 1.2.3 FONT

Über den Befehl **font** kann die Schriftart und die Schriftgröße für das Diagramm definiert werden. Die Schrift wird durch die Parameter **fontname**, **fontsize** und **fontstyle** definiert. Die Parameter **fontsize** und **fontstyle** sind optionale Parameter und müssen nicht angegeben werden. In diesem Fall werden die Standardwerte verwendet. Die definierte Schrift wird für alle Texte innerhalb des Diagramms verwendet.

➤ **Syntax**

**font:** fontname, fontsize, fontstyle

➤ **Parameter**

- fontname

Der Wert dieses Parameters muss der Name einer auf dem System vorhandenen Schriftart sein.

Der Standardwert für den Parameter ist "Arial".

- `fontsize` (optional)

Dieser Parameter definiert die Größe der Schrift in Pixel. Die Schriftgröße kann zwischen 6 und 20 Pixel betragen. Bei der Eingabe von Kommazahlen muss beachtet werden, dass diese in Anführungszeichen angegeben werden müssen. Der Standardwert für den Parameter ist "10".

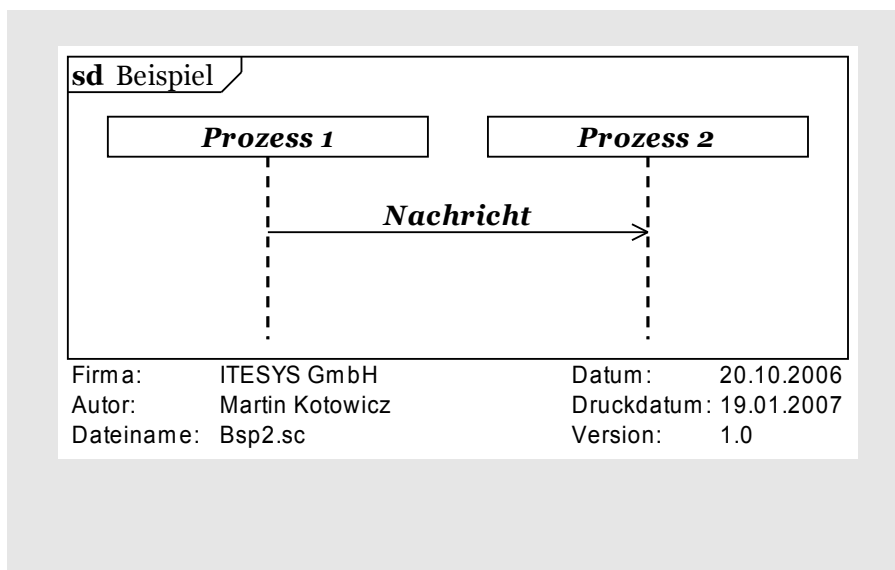
- `fontstyle` (optional)

Dieser Parameter definiert den Schnitt der Schrift. Es können mehrere Werte angegeben werden, diese müssen dann durch ein Leerzeichen voneinander getrennt sein.

Wert	Bedeutung
regular	Standard
bold	Fett
italic	Kursiv
strikeout	Durchgestrichen
underline	Unterstrichen

Bitte beachten Sie, dass nicht alle Schriftarten alle Schnitte unterstützen. Der Standardwert für den Parameter ist "regular".

### ➤ Beispiel



**DiagramName:** Beispiel  
**DiagramStyle:** uml

**Font:** "Georgia", "10,5", "Italic Bold"

**PrintAuthor:** yes  
**PrintCompany:** yes  
**PrintVersion:** yes  
**PrintDate:** yes  
**PrintCreationDate:** yes  
**PrintFileName:** yes  
**PageMargins:** 5, 5, 5, 5  
**PrintFootLine:** yes  
**Author:** Martin Kotowicz  
**Company:** ITESYS GmbH  
**Version:** 1.0  
**Date:** 20.10.2006  
**PageSize:** 400, auto

**process:** p1, Prozess 1  
**process:** p2, Prozess 2  
**msg:** p1, p2, Nachricht

### 1.2.4 LEFT

Über den Befehl **left** kann zusätzlicher Rand links von der ersten Instanz definiert werden. Dieser Rand kann z.B. für das Platzieren von Kommentaren verwendet werden.

### ➤ Syntax

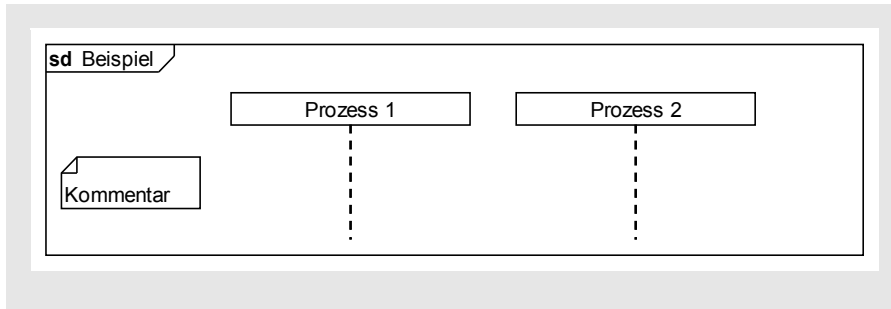
**left:** width

➤ **Parameter**

- width

Die Breite des Randes. Die Eingabe kann nur in Bildpunkten (Pixel) erfolgen.

➤ **Beispiel**



**DiagramName:** Beispiel  
**DiagramStyle:** uml  
**PageSize:** auto, auto  
**PageMargins:** 10,10,10,10  
**Left:** 100  
**Right:** 50

**process:** p1, Prozess 1  
**process:** p2, Prozess 2

**comment:** ENV\_LEFT, Kommentar

### 1.2.5 RIGHT

Über den Befehl **right** kann zusätzlicher Rand rechts von der letzten Instanz definiert werden. Dieser Rand kann z.B. für das Platzieren von Kommentaren verwendet werden.

➤ **Syntax**

**right:** width

➤ **Parameter**

- width

Die Breite des Randes. Die Eingabe kann nur in Bildpunkten (Pixel) erfolgen.

### 1.2.6 LINEOFFSET

Über den Befehl **lineoffset** kann die Höhe der Diagrammzeilenzwischenräume definiert werden.

➤ **Syntax**

**lineoffset:** height

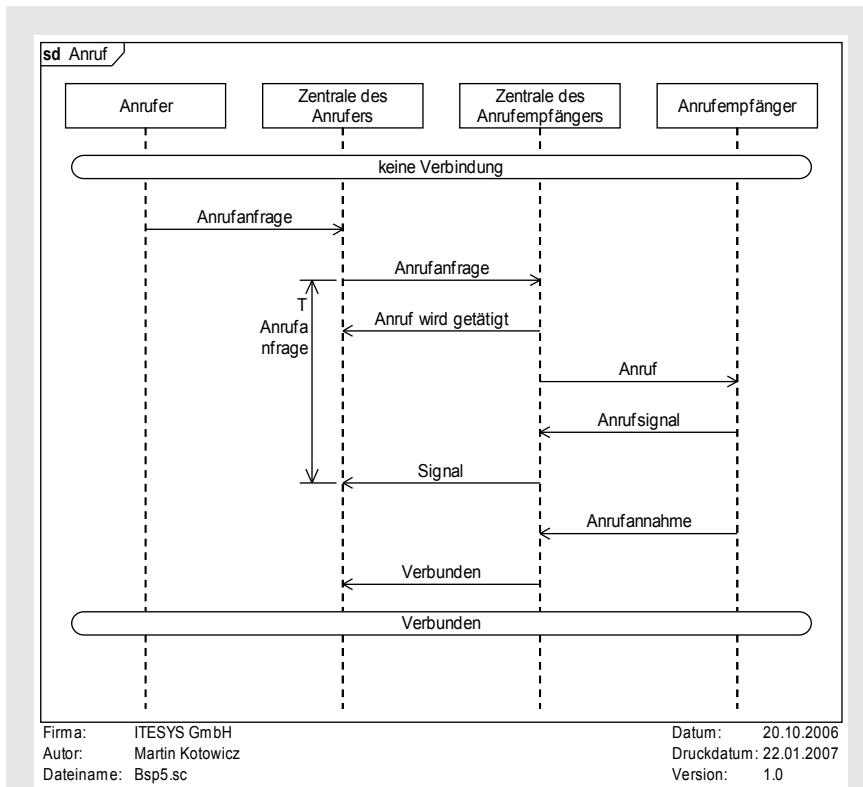
➤ **Parameter**

- height

Die Höhe des Zeilenzwischenraumes im Diagramm. Der Wert des Parameters muss zwischen "1" und "100" Bildpunkten (Pixel) liegen. Die Eingabe in einer anderen Größeneinheit ist nicht möglich.

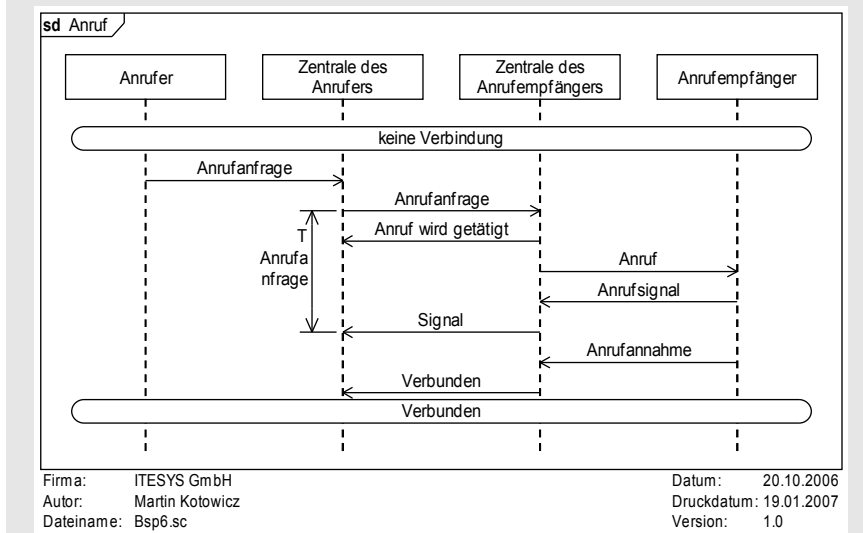
Der Standardwert für den Parameter ist "20".

➤ Beispiel



**PrintAuthor:** yes  
**PrintCompany:** yes  
**PrintVersion:** yes  
**PrintDate:** yes  
**PrintCreationDate:** yes  
**PrintFileName:** yes  
**Font:** "Arial", "10", "Regular"  
**DiagramName:** Anruf  
**DiagramStyle:** uml  
**PageMargins:** 5, 5, 5, 5  
**PrintFootLine:** yes  
**Author:** Martin Kotowicz  
**Company:** ITESYS GmbH  
**Version:** 1.0  
**Date:** 20.10.2006  
**PageSize:** 660, auto

**process:** CLU, "Anrufer"  
**process:** ON, "Zentrale des Anrufers"  
**process:** DN, "Zentrale des Anrufempfängers"  
**process:** CDU, "Anrufempfänger"  
**stateoverall:** "keine Verbindung";  
**msg:** CLU, ON, "Anrufanfrage";  
**timerbegin:** t1, ON, T Anrufanfrage  
**msg:** ON, DN, Anrufanfrage;  
**msg:** DN, ON, Anruf wird getätigt;  
**msg:** DN, CDU, Anruf;  
**msg:** CDU, DN, Anrufsignal;  
**msg:** DN, ON, Signal  
**timerend:** t1;  
**msg:** CDU, DN, Anrufannahme;  
**msg:** DN, ON, Verbunden;  
**stateoverall:** Verbunden;



**PrintAuthor:** yes  
**PrintCompany:** yes  
**PrintVersion:** yes  
**PrintDate:** yes  
**PrintCreationDate:** yes  
**PrintFileName:** yes  
**Font:** "Arial", "10", "Regular"  
**DiagramName:** Anruf  
**DiagramStyle:** uml  
**PageMargins:** 5, 5, 5, 5  
**PrintFootLine:** yes  
**Author:** Martin Kotowicz  
**Company:** ITESYS GmbH  
**Version:** 1.0  
**Date:** 20.10.2006  
**PageSize:** 660, auto

**Lineoffset:** 5

**process:** CLU, "Anrufer"  
**process:** ON, "Zentrale des Anrufers"  
**process:** DN, "Zentrale des Anrufempfängers"  
**process:** CDU, "Anrufempfänger"  
**stateoverall:** "keine Verbindung";  
**msg:** CLU, ON, "Anrufanfrage";  
**timerbegin:** t1, ON, T Anrufanfrage  
**msg:** ON, DN, Anrufanfrage;  
**msg:** DN, ON, Anruf wird getätigt;  
**msg:** DN, CDU, Anruf;  
**msg:** CDU, DN, Anrufsignal;  
**msg:** DN, ON, Signal  
**timerend:** t1;  
**msg:** CDU, DN, Anrufannahme;  
**msg:** DN, ON, Verbunden;  
**stateoverall:** Verbunden;

## 1.2.7 NEXTPAGE

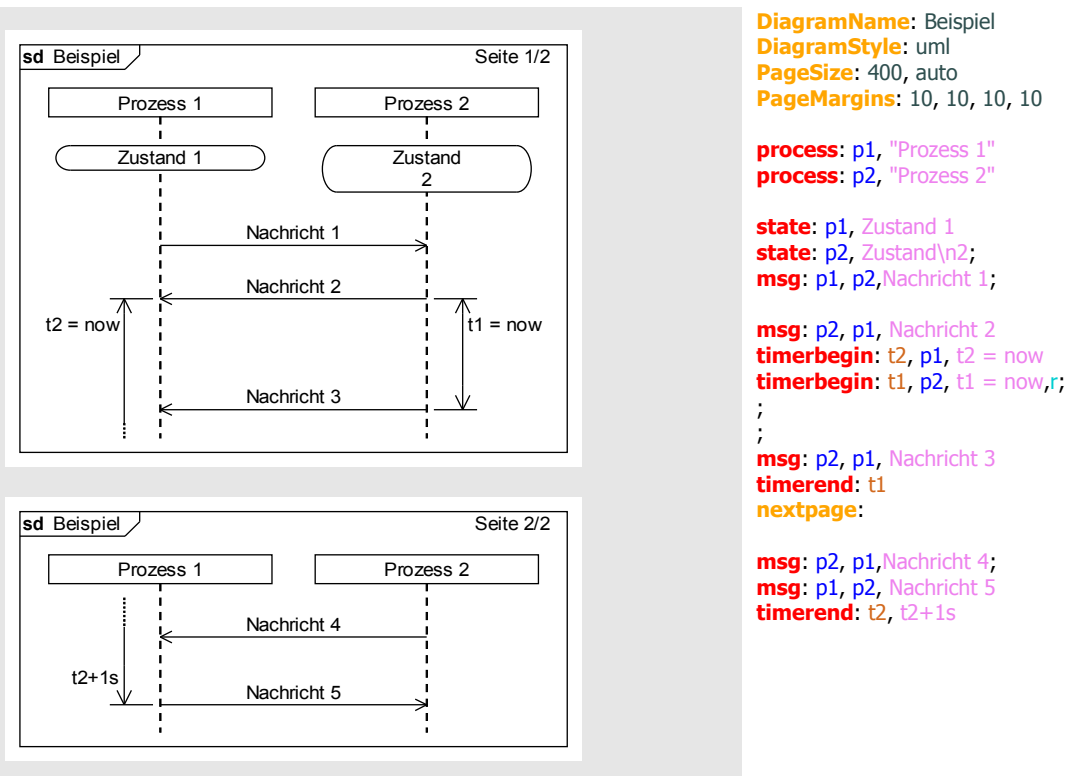
Über den Befehl **nextpage** wird ein manueller Seitenumbruch definiert.

### ➤ Syntax

**nextpage:**

### ➤ Parameter

### ➤ Beispiel



## 1.3 Fußzeile

Die Fußzeile des Diagramms kann beim Ausdrucken und beim Speichern des Diagramms in eine Grafikdatei ausgegeben werden. Sie beinhaltet Zusatzinformationen zum Diagramm wie z.B. das Erstellungsdatum und die Version.

### 1.3.1 PRINTFOOTLINE

Über den Befehl **printfootline** kann die Ausgabe der Fußzeile des Diagramms beim Drucken und Exportieren ein- und ausgeschaltet werden. Die Ausgabe wird über den Parameter **option** gesteuert.

➤ **Syntax**

**printfootline**: option

➤ **Parameter**

- option

Wert	Bedeutung
yes	Die Fußzeile wird ausgegeben.
no	Die Fußzeile wird nicht ausgegeben.

Der Standardwert für den Parameter ist "no".

### 1.3.2 AUTHOR

Über den Befehl **author** kann der Name des Autors des Diagramms definiert werden. Der Autorname wird durch den Parameter `name` festgelegt und erscheint beim Drucken und Exportieren des Diagramms in der Fußzeile, vorausgesetzt die Parameter der Befehle **printauthor** und **printfootline** haben den Wert "yes".

➤ **Syntax**

**author**: name

➤ **Parameter**

- name

Der Name des Diagrammautors. Dieser Parameter darf einen beliebigen alphanumerischen Wert annehmen.

### 1.3.3 PRINTAUTHOR

Über den Befehl **printauthor** kann die Ausgabe des durch den Befehl **author** definierten Autors in der Fußzeile beim Drucken und Exportieren ein- und ausgeschaltet werden. Die Ausgabe wird über den Parameter `option` gesteuert.

**Wichtig:** Die Ausgabe des Autors in der Fußzeile ist zusätzlich von dem Befehl **printfootline** abhängig.

➤ **Syntax**

**printauthor**: option

➤ **Parameter**

- option

Wert	Bedeutung
yes	Der Autor des Diagrams wird ausgegeben.
no	Der Autor des Diagrams wird nicht ausgegeben.

Der Standardwert für den Parameter ist "yes".

### 1.3.4 COMPANY

Über den Befehl **company** kann der Name der Firma definiert werden. Der Firmenname wird durch den Parameter `name` festgelegt und erscheint beim Drucken und Exportieren des Diagramms in der Fußzeile, vorausgesetzt die Parameter der Befehle **printcompany** und **printfootline** haben den Wert "yes".

➤ **Syntax**

**company**: name

➤ **Parameter**

- name

Der Name der Firma. Dieser Parameter darf einen beliebigen alphanumerischen Wert annehmen.

### 1.3.5 PRINTCOMPANY

Über den Befehl **printcompany** kann die Ausgabe der durch den Befehl **company** definierten Firma in der Fußzeile beim Drucken und Exportieren ein- und ausgeschaltet werden. Die Ausgabe wird über den Parameter `option` gesteuert.

**Wichtig:** Die Ausgabe der Firma in der Fußzeile ist zusätzlich von dem Befehl **printfootline** abhängig.

➤ **Syntax**

**printcompany**: option

➤ **Parameter**

- option

Wert	Bedeutung
yes	Die Firma wird ausgegeben.
no	Die Firma wird nicht ausgegeben.

Der Standardwert für den Parameter ist "yes".

### 1.3.6 PRINTFILENAME

Über den Befehl **printfilename** kann die Ausgabe des Dateinamens in der Fußzeile beim Drucken und Exportieren ein- und ausgeschaltet werden. Die Ausgabe wird über den Parameter **option** gesteuert.

**Wichtig:** Die Ausgabe des Dateinamens in der Fußzeile ist zusätzlich von dem Befehl **printfootline** abhängig.

➤ **Syntax**

**printfilename:** option

➤ **Parameter**

- option

Wert	Bedeutung
yes	Der Dateiname wird ausgegeben.
no	Der Dateiname wird nicht ausgegeben.

Der Standardwert für den Parameter ist "yes".

### 1.3.7 DATE

Über den Befehl **date** kann ein Datum definiert werden. Das Datum wird durch den Parameter **date** festgelegt und erscheint beim Drucken und Exportieren des Diagramms in der Fußzeile, vorausgesetzt die Parameter der Befehle **printdate** und **printfootline** haben den Wert "yes".

➤ **Syntax**

**date:** date

➤ **Parameter**

- date

Das Datum. Dieser Parameter darf einen beliebigen alphanumerischen Wert annehmen.



### 1.3.8 PRINTDATE

Über den Befehl **printdate** kann die Ausgabe des durch den Befehl **date** definierten Datums in der Fußzeile beim Drucken und Exportieren ein- und ausgeschaltet werden. Die Ausgabe wird über den Parameter **option** gesteuert.

**Wichtig:** Die Ausgabe des Datums in der Fußzeile ist zusätzlich von dem Befehl **printfootline** abhängig.

➤ **Syntax**

**printdate**: option

➤ **Parameter**

- option

Wert	Bedeutung
yes	Das Datum wird ausgegeben.
no	Das Datum wird nicht ausgegeben.

Der Standardwert für den Parameter ist "yes".

### 1.3.9 PRINTCREATIONDATE

Über den Befehl **printcreationdate** kann die Ausgabe des aktuellen Datums in der Fußzeile beim Drucken und Exportieren ein- und ausgeschaltet werden. Die Ausgabe wird über den Parameter **option** gesteuert.

**Wichtig:** Die Ausgabe des Datums in der Fußzeile ist zusätzlich von dem Befehl **printfootline** abhängig.

➤ **Syntax**

**printcreationdate**: option

➤ **Parameter**

- option

Wert	Bedeutung
yes	Das aktuelle Datum wird ausgegeben.
no	Das aktuelle Datum wird nicht ausgegeben.

Der Standardwert für den Parameter ist "yes".

### 1.3.10VERSION

Über den Befehl **version** kann die Versionsnummer des Diagramms definiert werden. Die Versionsnummer wird durch den Parameter `number` festgelegt und erscheint beim Drucken und Exportieren des Diagramms in der Fußzeile, vorausgesetzt die Parameter der Befehle **printversion** und **printfootline** haben den Wert "yes".

➤ **Syntax**

**version**: number

➤ **Parameter**

- number

Die Versionsnummer des Diagramms. Dieser Parameter darf einen beliebigen alphanumerischen Wert annehmen.

### 1.3.11PRINTVERSION

Über den Befehl **printversion** kann die Ausgabe der durch den Befehl **version** definierten Versionsnummer in der Fußzeile beim Drucken und Exportieren ein- und ausgeschaltet werden. Die Ausgabe wird über den Parameter `option` gesteuert.

**Wichtig:** Die Ausgabe der Versionsnummer in der Fußzeile ist zusätzlich von dem Befehl **printfootline** abhängig.

➤ **Syntax**

**printversion**: option

➤ **Parameter**

- option

Wert	Bedeutung
yes	Die Versionsnummer wird ausgegeben.
no	Die Versionsnummer wird nicht ausgegeben.

Der Standardwert für den Parameter ist "yes".

## 1.4 Farbdefinition

Die Darstellung der Diagrammelemente kann farblich verändert werden. Dies betrifft sowohl die Hintergrundfarbe der Elemente als auch die Textfarbe dieser.

## 1.4.1 BACKCOLOR

Über den Befehl **backcolor** kann die Texthintergrundfarbe der nachfolgenden Diagrammelemente festgelegt werden. Die Texthintergrundfarbe wird durch den Parameter **color** festgelegt und ist bis zum nächsten **backcolor** - Befehl gültig. Die Texthintergrundfarbe wirkt sich auf alle Diagrammelemente, die nicht durch einen geschlossenen Rahmen dargestellt werden. Für diese Elemente muss der Befehl **fillcolor** verwendet werden.

### ➤ Syntax

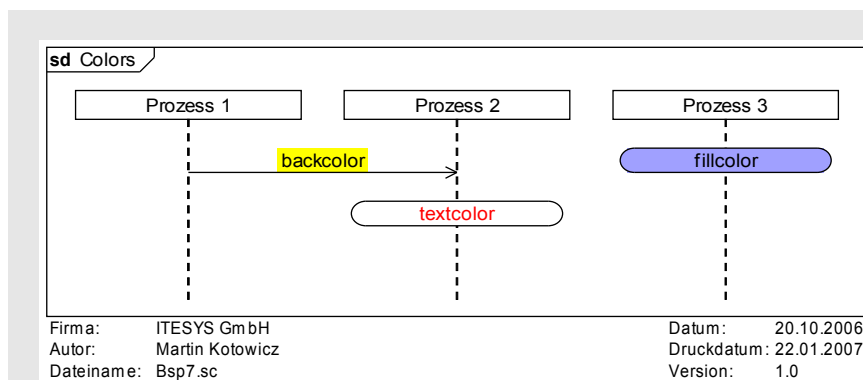
**backcolor**: color

### ➤ Parameter

- color

Die Farbe, in der der Texthintergrund dargestellt werden soll. Die Farbe muss im HTML-Format angegeben werden, z.B.: "red", "darkgreen", "#FF22A0". Durch die Eingabe von "none" wird der Standardwert "white" verwendet. Eine Liste aller gültigen Farbnamen befindet sich im Anhang A dieses Handbuchs.

### ➤ Beispiel



**PrintAuthor**: yes  
**PrintCompany**: yes  
**PrintVersion**: yes  
**PrintDate**: yes  
**PrintCreationDate**: yes  
**PrintFileName**: yes  
**Font**: "Arial", "10", "Regular"  
**DiagramName**: Colors  
**DiagramStyle**: uml  
**PageMargins**: 5, 5, 5, 5  
**PrintFootLine**: yes  
**Author**: Martin Kotowicz  
**Company**: ITESYS GmbH  
**Version**: 1.0  
**Date**: 20.10.2006  
**PageSize**: auto, auto  
**process**: P1, "Prozess 1"  
**process**: P2, "Prozess 2"  
**process**: P3, "Prozess 3"  
**backcolor**: yellow  
**fillcolor**: #A0A0FF  
**msg**: P1,P2, backcolor  
**state**: P3, fillcolor;  
**backcolor**: none  
**fillcolor**:none  
**textcolor**: red  
**state**: P2, textcolor;

## 1.4.2 FILLCOLOR

Über den Befehl **fillcolor** kann die Hintergrundfarbe der nachfolgenden, durch einen Rahmen geschlossenen, Diagrammelemente festgelegt werden. Die Hintergrundfarbe wird durch den Parameter **color** festgelegt und ist bis zum nächsten **fillcolor** - Befehl gültig. Diagrammelemente, die nicht durch einen geschlossenen Rahmen dargestellt werden, werden durch diesen Befehl nicht verändert. Für diese Elemente muss der Befehl **backcolor** verwendet werden.

➤ **Syntax**

**fillcolor**: color

➤ **Parameter**

- color

Die Farbe, in der der Hintergrund dargestellt werden soll. Die Farbe muss im HTML-Format angegeben werden, z.B.: "red", "darkgreen", "#FF22A0". Durch die Eingabe von "none" wird der Standardwert "white" verwendet. Eine Liste aller gültigen Farbnamen befindet sich im Anhang A dieses Handbuchs.

### 1.4.3 TEXTCOLOR

Über den Befehl **textcolor** kann die Textfarbe der nachfolgenden Diagrammelemente festgelegt werden. Die Textfarbe wird durch den Parameter `color` festgelegt und ist bis zum nächsten **textcolor** - Befehl gültig.

➤ **Syntax**

**textcolor**: color

➤ **Parameter**

- color

Die Farbe, in der der Text der Diagrammelemente dargestellt werden soll. Die Farbe muss im HTML-Format angegeben werden, z.B.: "red", "darkgreen", "#FF22A0". Durch die Eingabe von "none" wird der Standardwert "black" verwendet. Eine Liste aller gültigen Farbnamen befindet sich im Anhang A dieses Handbuchs.

## 2 Diagrammelemente

Diagrammelemente sind alle im Sequenzdiagramm dargestellten Elemente. Dazu zählen unter Anderem Lebenslinien, Nachrichten, Zustände, kombinierte Fragmente. Dieses Kapitel listet die Befehle der Diagrammbeschreibung auf, mit denen diese Elemente erzeugt werden.

### 2.1 Lebenslinie

Alle Lebenslinien müssen zuerst durch anlegen von Instanzen definiert werden, bevor sie im Sequenzdiagramm verfügbar sind. Die Definition einer Instanz ist von ihrer gewünschten Darstellungsform abhängig. Es gibt zwei mögliche Darstellungsformen: Akteur und Prozess. Unter Prozess sind dabei alle „Classifier“, wie z.B. „Objekte“ und „Klassen“ zusammengefasst. Im Folgenden wird einheitlich der Begriff „Prozess“ verwendet. Die Darstellung der Lebenslinie kann ebenfalls verändert werden.

#### 2.1.1 ACTOR

Über den Befehl **actor** kann in das Diagramm ein Akteur hinzugefügt werden. Ein Akteur stellt eine Rolle im Kontext eines modellierten Systems dar, die auf das System einwirkt oder von diesem beeinflusst wird. Ein Akteur kann einen Menschen, eine Gruppe von Menschen, einen Sensor oder Akteur, oder eine Applikation darstellen. Über den Parameter **id** wird dem Akteur ein eindeutiger Identifikator seiner Lebenslinie zugewiesen. Über die beiden Parameter **name** und **description** wird der Name des Akteurs und optional eine Beschreibung festgelegt. Akteure und Prozesse werden horizontal gleichmäßig im Diagramm verteilt. Es kann aber notwendig sein, dass ein Akteur zu seinen Nachbarn eine größere Distanz einhält, als die vorgegebene. Mit den Parametern **left** und **right** kann zusätzlicher Platz links und rechts des Akteurs festgelegt werden.

##### ➤ Syntax

**actor**: **id**, **name**, **description**, **left**, **right**

##### ➤ Parameter

- **id**

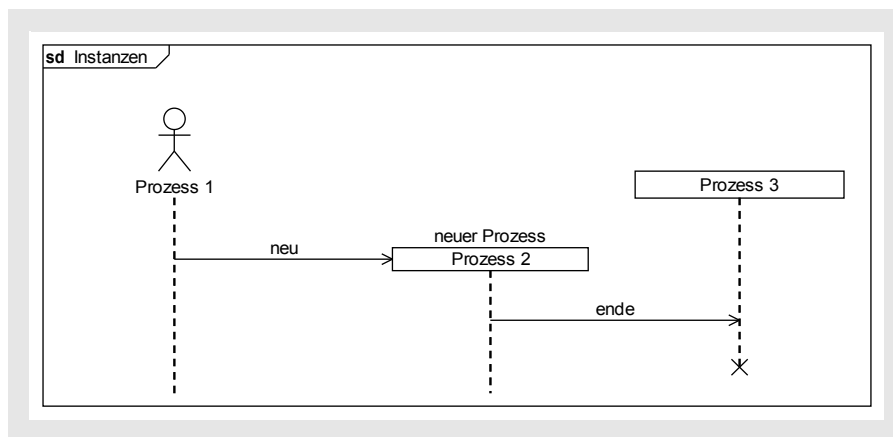
Der Identifikator der Lebenslinie. Ein beliebiger alphanumerischer Wert, der im gesamten Diagramm die Lebenslinie des Akteurs eindeutig identifiziert. Der Identifikator ist im Diagramm nicht sichtbar.

- **name**

Der Name des Akteurs. Ein beliebiger Text, der den Namen des Akteurs festlegt. Der Name erscheint im Diagramm unter dem Akteur. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.

- **description** (optional)  
Die Beschreibung des Akteurs. Ein beliebiger Text, der den Akteur beschreibt. Die Beschreibung erscheint im Diagramm über dem Akteur. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.
- **left** (optional)  
Zusätzlicher Platz links vom Akteur. Die Eingabe kann nur in Bildpunkten (Pixel) erfolgen.
- **right** (optional)  
Zusätzlicher Platz rechts vom Akteur. Die Eingabe kann nur in Bildpunkten (Pixel) erfolgen.

### ➤ Beispiel



**DiagramName:** Instanzen  
**DiagramStyle:** uml  
**PageSize:** auto  
**PageMargins:** 10,10,10,10

**actor:** p1, Prozess 1  
**dummyprocess:** p2, 50,0  
**process:** p3, "Prozess 3"

**create:** p1, p2, neu, Prozess 2, neuer Prozess;  
**msg:** p2, p3, ende;  
**stop:** p3

## 2.1.2 PROCESS

Über den Befehl **process** kann in das Diagramm ein Prozess (oder Objekt) hinzugefügt werden. Über den Parameter **id** wird dem Prozess ein eindeutiger Identifikator seiner Lebenslinie zugewiesen. Über die beiden Parameter **name** und **description** wird der Prozessname und optional eine Beschreibung festgelegt. Prozesse und Akteure werden horizontal gleichmäßig im Diagramm verteilt. Es kann aber notwendig sein, dass ein Objekt zu seinen Nachbarn eine größere Distanz einhält, als die vorgegebene. Mit den Parametern **left** und **right** kann zusätzlicher Platz links und rechts des Prozesses festgelegt werden.

### ➤ Syntax

**process:** id, name, description, left, right

### ➤ Parameter

- **id**  
Der Identifikator der Lebenslinie. Ein beliebiger alphanumerischer Wert, der im gesamten Diagramm die Lebenslinie des Prozesses eindeutig identifiziert. Der Identifikator ist im Diagramm nicht sichtbar.

- **name**  
Der Prozessname. Ein beliebiger Text, der den Namen des Prozesses festlegt. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.
- **description** (optional)  
Die Beschreibung des Prozesses. Ein beliebiger Text, der den Prozess beschreibt. Die Beschreibung erscheint im Diagramm über dem Prozess. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.
- **left** (optional)  
Zusätzlicher Platz links vom Prozess. Die Eingabe kann nur in Bildpunkten (Pixel) erfolgen.
- **right** (optional)  
Zusätzlicher Platz rechts vom Prozess. Die Eingabe kann nur in Bildpunkten (Pixel) erfolgen.

### 2.1.3 DUMMYPROCESS

Über den Befehl **dummyprocess** kann im Diagramm ein Platzhalter für einen Prozess, der erst später im Diagrammverlauf erstellt wird, angelegt werden. Ein Platzhalter-Prozess ist im Diagramm nicht sichtbar. Über den Parameter **id** wird dem Prozess ein eindeutiger Identifikator zugewiesen. Prozesse und Akteure werden horizontal gleichmäßig im Diagramm verteilt. Es kann aber notwendig sein, dass ein Objekt zu seinen Nachbarn eine größere Distanz einhält, als die vorgegebene. Mit den Parametern **left** und **right** kann zusätzlicher Platz links und rechts des Prozesses festgelegt werden.

#### ➤ **Syntax**

**dummyprocess**: **id**, **left**, **right**

#### ➤ **Parameter**

- **id**  
Der Identifikator der Lebenslinie. Ein beliebiger alphanumerischer Wert, der im gesamten Diagramm den Prozess eindeutig identifiziert. Der Identifikator ist im Diagramm nicht sichtbar.
- **left** (optional)  
Zusätzlicher Platz links vom Prozess. Die Eingabe kann nur in Bildpunkten (Pixel) erfolgen.

- **right** (optional)

Zusätzlicher Platz rechts vom Prozess. Die Eingabe kann nur in Bildpunkten (Pixel) erfolgen.

#### 2.1.4 CREATE

Über den Befehl **create** kann eine Erzeugungsnachricht hinzugefügt werden. Die Lebenslinie des so erzeugten Prozesses muss vorher mit dem Befehl **dummyprocess** reserviert worden sein.

##### ➤ Syntax

**create**: **sourceid**, **destinationid**, **message**, **name**, **description**

##### ➤ Parameter

- **sourceid**

Der Identifikator der Lebenslinie, von der die Erzeugungsnachricht gesendet wird.

- **destinationid**

Der Identifikator der zuvor mit dem Befehl **dummyprocess** reservierten Lebenslinie, an der der Prozess erstellt werden soll.

- **message**

Ein beliebiger Text, der auf der erzeugenden Nachricht erscheinen soll. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.

- **name**

Der Prozessname. Ein beliebiger Text, der den Namen des Prozesses festlegt. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.

- **description** (optional)

Die Beschreibung des Prozesses. Ein beliebiger Text, der den Prozess beschreibt. Die Beschreibung erscheint im Diagramm über dem Prozess. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.

#### 2.1.5 STOP

Über den Befehl **stop** kann ein Stop-Symbol zu einer Lebenslinie hinzugefügt werden. Die Lebenslinie wird ab dieser Stelle nicht mehr gezeichnet.



➤ **Syntax**

**stop**: `instanceid`

➤ **Parameter**

- `instanceid`

Der Identifikator der Lebenslinie, die mit dem Stop-Symbol beendet werden soll.

## 2.1.6 REGIONBEGIN

Über die Befehle **regionbegin** und **regionend** kann die Darstellung der Lebenslinie verändert werden. Der Befehle **regionbegin** definiert dabei den Anfang der geänderten Darstellung. Die Lebenslinie einer Instanz kann als Aktivität, Coregion oder Suspension dargestellt werden.

➤ **Syntax**

**regionbegin**: `instanceid`, `style`

➤ **Parameter**

- `instanceid`

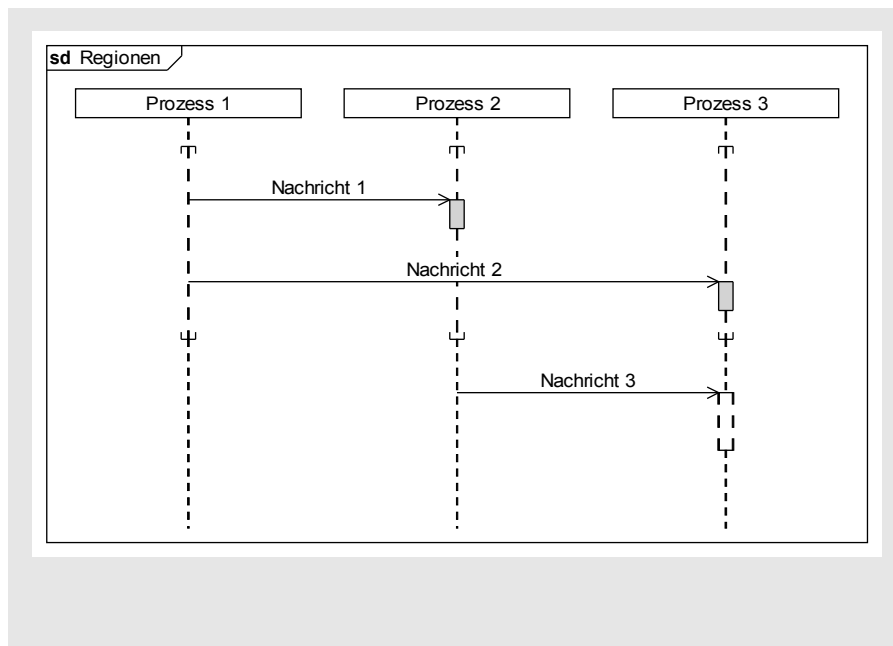
Der Identifikator der Lebenslinie, für die die Darstellung geändert werden soll.

- `style`

Legt die neue Darstellungsform der Lebenslinie fest.

Wert	Bedeutung
activation	Die Lebenslinie wird als Aktivität dargestellt
coregion	Die Lebenslinie wird als Coregion dargestellt
suspension	Die Lebenslinie wird als Suspension dargestellt

➤ **Beispiel**



**DiagramName:** Regionen  
**DiagramStyle:** uml  
**PageSize:** auto, auto  
**PageMargins:** 10, 10, 10, 10

**process:** p1, "Prozess 1"  
**process:** p2, "Prozess 2"  
**process:** p3, "Prozess 3"

**regionbegin:** p1, coregion  
**regionbegin:** p2, coregion  
**regionbegin:** p3, coregion;  
**msg:** p1,p2, Nachricht 1  
**regionbegin:** p2, activation;  
**regionend:** p2;  
**regionbegin:** p3, activation  
**msg:** p1,p3, Nachricht 2;  
**regionend:** p3;  
**regionend:** p1  
**regionend:** p2  
**regionend:** p3;  
**msg:** p2,p3, Nachricht 3  
**regionbegin:** p3, suspension;  
;  
**regionend:** p3;

## 2.1.7 REGIONEND

Über die Befehle **regionbegin** und **regionend** kann die Darstellung der Lebenslinie verändert werden. Der Befehle **regionend** definiert dabei das Ende der zuletzt geänderten und gültigen Darstellung. Die Lebenslinie einer Instanz kann als Aktivität, Coregion oder Suspension dargestellt werden.

➤ **Syntax**

**regionend:** instanceid

➤ **Parameter**

- instanceid

Der Identifikator der Lebenslinie, für die die geänderte Darstellung beendet werden soll. Die Lebenslinie wird danach in der zuletzt gültigen Form dargestellt.

## 2.2 Nachrichten

Nachrichten sind neben den Instanzen ein Hauptbestandteil von MSCs/SDs. Signale, Ereignisse, Befehle und Rückgabewerte werden hier unter dem Begriff Nachrichten zusammengefasst.

## 2.2.1 MSG

Über den Befehl **msg** kann eine Nachricht (engl. Message) im Diagramm dargestellt werden. Eine Nachricht definiert die Kommunikation und deren Richtung zwischen den Prozessen und Akteuren des Diagramms. Es können drei Nachrichtenarten im Diagramm dargestellt werden: asynchrone Nachrichten, synchrone Nachrichten und Antwortnachrichten, wobei in der SDL-Darstellung kein Unterschied zwischen asynchronen und synchronen Nachrichten gemacht wird. Soll die Nachricht einen zeitlichen (vertikalen) Versatz zwischen dem Sende- und Empfangszeitpunkt haben, so müssen die Befehle **msgbegin** und **msgend** verwendet werden.

### ➤ Syntax

**msg**: *sourceid*, *destinationid*, *message*, *style*

### ➤ Parameter

- **sourceid**

Der Identifikator der Lebenslinie der Instanz, die die Nachricht versendet oder eine der folgenden Konstanten:

Wert	Bedeutung
ENV_LEFT	Die Nachricht kommt von außerhalb des Diagramms von links.
ENV_RIGHT	Die Nachricht kommt von außerhalb des Diagramms von rechts.

- **destinationid**

Der Identifikator der Lebenslinie der Instanz, die die Nachricht empfängt oder eine der folgenden Konstanten:

Wert	Bedeutung
ENV_LEFT	Die Nachricht wird links außerhalb des Diagramms empfangen.
ENV_RIGHT	Die Nachricht wird rechts außerhalb des Diagramms empfangen.

- **message**

Ein beliebiger Text, der auf der Nachricht erscheinen soll. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.

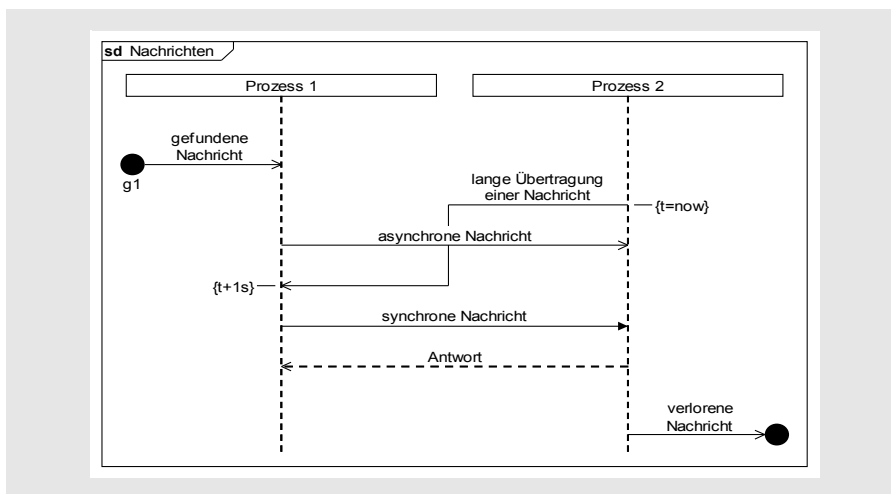
- **style** (optional)

Legt die Art der Nachricht fest.

Wert	Bedeutung
+	asynchrone Nachricht
!	synchrone Nachricht (in SDL-Diagrammen ohne Funktion)
*	Antwortnachricht

Der Standardwert für den Parameter ist "+".

### ➤ Beispiel



**DiagramName:** Nachrichten  
**DiagramStyle:** uml  
**PageSize:** 600, auto  
**PageMargins:** 10, 10, 10, 10

**process:** p1, "Prozess 1"  
**process:** p2, "Prozess 2"

**found:** p1, "gefundene Nachricht", "g1", !;  
**msgbegin:** m1, p2, p1, "lange Übertragung einer Nachricht"  
**linecomment:** p2, "{t=now}", r;  
**msg:** p1,p2, "asynchrone Nachricht";  
**msgend:** m1  
**linecomment:** p1, "{t+1s}";  
**msg:** p1,p2, "synchrone Nachricht", !;  
**msg:** p2, p1, "Antwort", \*;  
**lost:** p2, "verlorene Nachricht", , r

## 2.2.2 MSGBEGIN

Über die Befehle **msgbegin** und **msgend** kann eine Nachricht (engl. Message), deren Send- und Empfangszeitpunkt zeitlich (vertikal) versetzt liegt, im Diagramm dargestellt werden. Eine Nachricht definiert die Kommunikation und deren Richtung zwischen den Prozessen und Akteuren des Diagramms. Es können drei Nachrichtenarten im Diagramm dargestellt werden: asynchrone Nachrichten, synchrone Nachrichten und Antwortnachrichten, wobei in der SDL-Darstellung kein Unterschied zwischen asynchronen und synchronen Nachrichten gemacht wird. Soll die Nachricht keinen zeitlichen Versatz zwischen dem Send- und Empfangszeitpunkt haben, so muss der Befehl **msg** verwendet werden.

### ➤ Syntax

**msgbegin:** messageid, sourceid, destinationid, message, style

### ➤ Parameter

- **messageid**

Der Identifikator der gesendeten Nachricht. Der Identifikator wird für den Befehl **msgend** benötigt, damit dieser der richtigen Nachricht zugeordnet werden kann.

- **sourceid**

Der Identifikator der Lebenslinie der Instanz, die die Nachricht versendet oder eine der folgenden Konstanten:

Wert	Bedeutung
ENV_LEFT	Die Nachricht kommt von außerhalb des Diagramms von links.
ENV_RIGHT	Die Nachricht kommt von außerhalb des Diagramms von rechts.

- **destinationid**

Der Identifikator der Lebenslinie der Instanz, die die Nachricht empfängt oder eine der folgenden Konstanten:

Wert	Bedeutung
ENV_LEFT	Die Nachricht wird links außerhalb des Diagramms empfangen.
ENV_RIGHT	Die Nachricht wird rechts außerhalb des Diagramms empfangen.

- **message**

Ein beliebiger Text, der auf der Nachricht erscheinen soll. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.

- **style** (optional)

Legt die Art der Nachricht fest.

Wert	Bedeutung
+	asynchrone Nachricht
!	synchrone Nachricht (in SDL-Diagrammen ohne Funktion)
*	Antwortnachricht

Der Standardwert für den Parameter ist "+".

### 2.2.3 MSGEND

Über die Befehle **msgbegin** und **msgend** kann eine Nachricht (engl. Message), deren Sende- und Empfangszeitpunkt zeitlich (vertikal) versetzt liegt, im Diagramm dargestellt werden. Eine Nachricht definiert die Kommunikation und deren Richtung zwischen den Prozessen und Akteuren des Diagramms. Es können drei Nachrichtenarten im Diagramm dargestellt werden: asynchrone Nachrichten, synchrone Nachrichten und Antwortnachrichten, wobei in der SDL-Darstellung kein Unterschied zwischen asynchronen und synchronen Nachrichten gemacht wird. Soll die Nachricht keinen zeitlichen Versatz zwischen dem Sende- und Empfangszeitpunkt haben, so muss der Befehl **msg** verwendet werden.

#### ➤ Syntax

**msgend**: messageid

#### ➤ Parameter

- messageid

Der Identifikator der Nachricht, die zu diesem Zeitpunkt ihren Empfänger erreichen soll

### 2.2.4 FOUND

Über den Befehl **found** kann eine gefundene Nachricht (engl. Found Message) im Diagramm dargestellt werden. Eine gefundene Nachricht ist eine Nachricht, dessen Absender unbekannt oder irrelevant ist und im Diagramm nicht dargestellt werden soll.

#### ➤ Syntax

**found**: instanceid, message, gate, orientation

#### ➤ Parameter

- instanceid

Der Identifikator der Lebenslinie, die die gefundene Nachricht empfangen soll.

- message

Ein beliebiger Text, der auf der gefundenen Nachricht erscheinen soll. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.

- gate (optional)

Ein beliebiger Text, der unter der gefundenen Nachricht erscheinen soll. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.

- **orientation** (optional)

Legt die Position der gefundenen Nachricht bezüglich der durch den Parameter **instanceid** definierten Lebenslinie fest.

Wert	Bedeutung
l	Die gefundene Nachricht wird links von der Lebenslinie dargestellt.
r	Die gefundene Nachricht wird rechts von der Lebenslinie dargestellt.

Der Standardwert für den Parameter ist "l".

### 2.2.5 LOST

Über den Befehl **lost** kann eine verlorene Nachricht (engl. Lost Message) im Diagramm dargestellt werden. Eine verlorene Nachricht ist eine Nachricht, deren Empfang nicht dargestellt wird weil der Empfänger zum Entwurfszeitpunkt unbekannt ist oder weil die Nachricht tatsächlich nie ankommt.

#### ➤ Syntax

**lost**: **instanceid**, **message**, **gate**, **orientation**

#### ➤ Parameter

- **instanceid**

Der Identifikator der Lebenslinie, die die verlorene Nachricht versendet.

- **message**

Ein beliebiger Text, der auf der verlorenen Nachricht erscheinen soll. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.

- **gate** (optional)

Ein beliebiger Text, der unter der verlorenen Nachricht erscheinen soll. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.

- **orientation** (optional)

Legt die Position der verlorenen Nachricht bezüglich der durch den Parameter **instanceid** definierten Lebenslinie fest.

Wert	Bedeutung
l	Die verlorene Nachricht wird links von der Lebenslinie dargestellt.
r	Die verlorene Nachricht wird rechts von der Lebenslinie dargestellt.

Der Standardwert für den Parameter ist "l".

## 2.3 Zustandsinvarianten

Die Zustandsinvarianten erlauben es im Sequenzdiagramm den Zustand einer Instanz zu spezifizieren.

### 2.3.1 STATE

Über den Befehl **state** können im Diagramm Zustandsinvarianten (engl. StateInvariant) modelliert werden. Die Zustandsinvarianten können an der Lebenslinie eines oder mehrerer Instanzen dargestellt werden. Soll sich die Zustandsinvariante über alle Instanzen erstrecken, so kann der Befehl **stateoverall** verwendet werden.

#### ➤ Syntax

**state**: *instanceid*, *text*, *style*, *orientation*

#### ➤ Parameter

- *instanceid*

Der Identifikator der Lebenslinie, für die die Zustandsinvariante modelliert werden soll. Soll die Zustandsinvariante für mehrere Lebenslinien modelliert werden, so müssen die Identifikatoren der Lebenslinien in Anführungszeichen, durch Komma getrennt, angegeben werden.

- *text*

Ein beliebiger Text, der den Zustand beschreibt. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.

- *style* (optional)

Legt die Darstellungsform der Zustandsinvariante fest.

Wert	Bedeutung
-	Die Zustandsinvariante wird durch ein Zustandsymbol dargestellt
*	Die Zustandsinvariante wird textuel in geschweiften Klammern dargestellt



Der Standardwert für den Parameter ist "-".

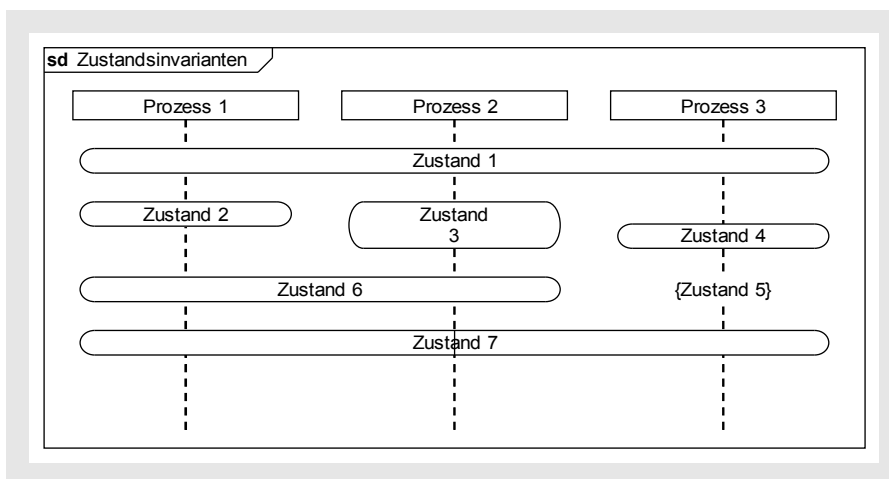
- **orientation** (optional)

Legt die vertikale Ausrichtung der Zustandsinvariante in der Diagrammzeile fest.

Wert	Bedeutung
t	Die Zustandsinvariante wird oben ausgerichtet
b	Die Zustandsinvariante wird unten ausgerichtet

Der Standardwert für den Parameter ist "t".

### ➤ Beispiel



**DiagramName:** Zustandsinvarianten

**DiagramStyle:** uml

**PageSize:** auto, auto

**PageMargins:** 10, 10, 10, 10

**process:** p1, "Prozess 1"

**process:** p2, "Prozess 2"

**process:** p3, "Prozess 3"

**stateoverall:** Zustand 1;

**state:** p1, Zustand 2,,t

**state:** p2, Zustand\n3

**state:** p3, Zustand 4,-,b;

**state:** p3, Zustand 5,\*,b

**state:** "p1,p2", Zustand 6;

**state:** "p1,p3", Zustand 7;

## 2.3.2 STATEOVERALL

Über den Befehl **stateoverall** können im Diagramm Zustandsinvarianten (engl. StateInvariant) über alle Instanzen modelliert werden. Soll sich die Zustandsinvariante nur über bestimmte Instanzen erstrecken, so kann der Befehl **state** verwendet werden.

### ➤ Syntax

**stateoverall:** text, orientation

### ➤ Parameter

- **text**

Ein beliebiger Text, der den Zustand beschreibt. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.

- **orientation** (optional)

Legt die vertikale Ausrichtung der Zustandsinvariante in der Diagrammzeile fest.

Wert	Bedeutung
t	Die Zustandsinvariante wird oben ausgerichtet
b	Die Zustandsinvariante wird unten ausgerichtet

Der Standardwert für den Parameter ist "t".

## 2.4 Tätigkeiten

Dieses SDL-spezifische Sequenzdiagrammelement erlaubt es die Aktionen zu definieren, die eine Instanz an einer Stelle des Sequenzdiagramms durchführt. Im Sequenzdiagrammgenerator ist dieses Element auch für die UML-Darstellung verfügbar.

### 2.4.1 TASK

Über den Befehl **task** können im Diagramm SDL-spezifische Aktivitäten definiert werden.

#### ➤ Syntax

**task**: *instanceid*, *text*, *orientation*

#### ➤ Parameter

- *instanceid*

Der Identifikator der Lebenslinie, für die die Aktivität modelliert werden soll.

- *text*

Ein beliebiger Text, der die Aktivität beschreibt. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.

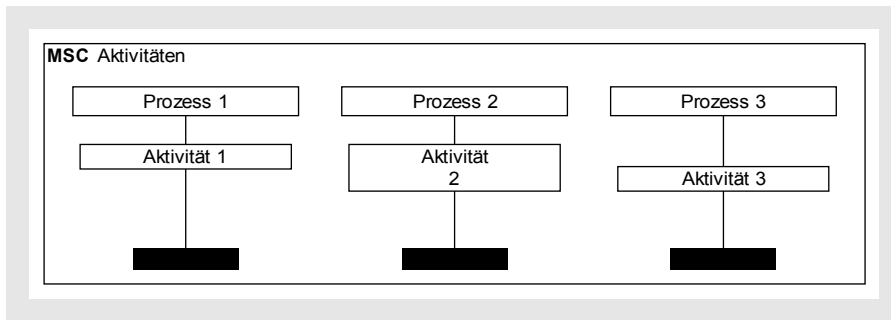
- *orientation* (optional)

Legt die vertikale Ausrichtung der Aktivität in der aktuellen Diagrammzeile fest.

Wert	Bedeutung
t	Die Aktivität wird oben ausgerichtet
b	Die Aktivität wird unten ausgerichtet

Der Standardwert für den Parameter ist "t".

➤ **Beispiel**



**DiagramName:** Aktivitäten  
**DiagramStyle:** sdl  
**PageSize:** auto, auto  
**PageMargins:** 10, 10, 10, 10

**process:** p1, "Prozess 1"  
**process:** p2, "Prozess 2"  
**process:** p3, "Prozess 3"

**task:** p1, Aktivität 1,t  
**task:** p2, Aktivität 2\n2  
**task:** p3, Aktivität 3,b;

## 2.5 Kommentare

Kommentare in Sequenzdiagrammen sind frei definierbare Texte, die in der Regel zusätzliche Informationen über die Sequenz enthalten.

### 2.5.1 COMMENT

Über den Befehl **comment** können im Diagramm Kommentare hinzugefügt werden. Die Kommentare können an einer Lebenslinie einer Instanz oder an den Seiten des Diagramms eingefügt werden. Um die Position des Kommentars festzulegen dienen die Parameter **instanceid** und **orientation**. Der Kommentartext wird in dem Parameter **text** definiert. Soll sich ein Kommentar über alle Instanzen erstrecken, so kann der Befehl **commentoverall** verwendet werden.

➤ **Syntax**

**comment:** instanceid, text, orientation

➤ **Parameter**

- **instanceid**

Der Identifikator der Lebenslinie, an der der Kommentar erscheinen soll oder eine der folgenden Konstanten:

Wert	Bedeutung
ENV_LEFT	Linke Seite des Sequenzdiagramms
ENV_RIGHT	Rechte Seite des Sequenzdiagramms

- **text**

Ein beliebiger Text, der als Kommentar erscheinen soll. Sollen im Kommentar Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.

- **orientation** (optional)

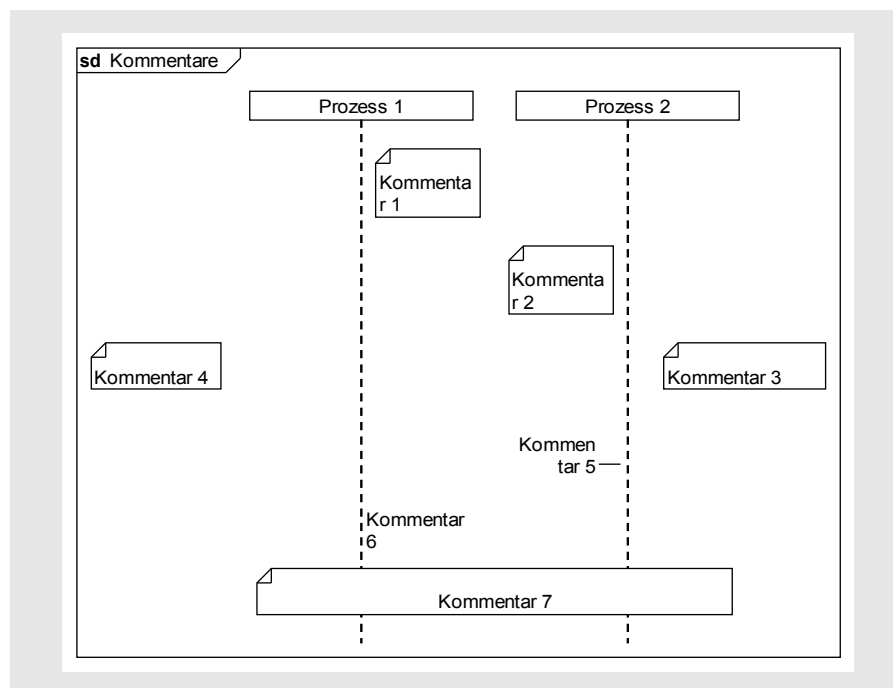
Legt die Position des Kommentars an der durch den Parameter **instanceid** definierten Lebenslinie.

Wert	Bedeutung
l	Der Kommentar wird links von der Lebenslinie dargestellt
r	Der Kommentar wird rechts von der Lebenslinie dargestellt

Für die Darstellung des Kommentars an den Seiten des Diagramms ist dieser Parameter ohne Bedeutung.

Der Standardwert für den Parameter ist "l".

### ➤ Beispiel



**DiagramName:** Kommentare

**DiagramStyle:** uml

**PageSize:** auto

**PageMargins:** 10,10,10,10

**left:** 100

**right:** 50

**process:** p1, Prozess 1

**process:** p2, Prozess 2

**comment:** p1, Kommentar 1, r;

**comment:** p2, Kommentar 2, l;

**comment:** ENV\_RIGHT, Kommentar 3

**comment:** ENV\_LEFT, Kommentar 4;

**linecomment:** p2, Kommentar 5;

**linecomment:** p1, Kommentar 6, r, \*;

**commentoverall:** Kommentar 7

## 2.5.2 COMMENTOVERALL

Über den Befehl **commentoverall** kann im Diagramm ein Kommentar, der sich über alle Lebenslinien legt, definiert werden. Der Kommentartext wird in dem Parameter **text** angegeben.

### ➤ Syntax

**commentoverall:** text

➤ **Parameter**

- **text**  
Ein beliebiger Text, der als Kommentar erscheinen soll. Sollen im Kommentar Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.

### 2.5.3 LINECOMMENT

Über den Befehl **linecomment** können im Diagramm rahmenlose Kommentare hinzugefügt werden. Die Kommentare werden an einer Lebenslinie einer Instanz eingefügt. Um die Position des Kommentars festzulegen dienen die Parameter **instanceid** und **orientation**. Der Kommentartext wird in dem Parameter **text** definiert. Die Darstellung kann optional ohne die waagerechte Linie, die den Kommentar mit der Lebenslinie verbindet, erfolgen.

➤ **Syntax**

**linecomment:** **instanceid**, **text**, **orientation**, **format**

➤ **Parameter**

- **instanceid**  
Der Identifikator der Lebenslinie, an der der Kommentar erscheinen soll.
- **text**  
Ein beliebiger Text, der als Kommentar erscheinen soll. Sollen im Kommentar Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.
- **orientation** (optional)  
Legt die Position des Kommentars an der durch den Parameter **instanceid** definierten Lebenslinie.

Wert	Bedeutung
l	Der Kommentar wird links von der Lebenslinie dargestellt
r	Der Kommentar wird rechts von der Lebenslinie dargestellt

Der Standardwert für den Parameter ist "l".

- **format** (optional)  
Legt fest, ob der Kommentar durch eine waagerechte Linie mit der Lebenslinie verbunden werden soll.

Wert	Bedeutung
*	Der Kommentar wird ohne die waagerechte Linie dargestellt
-	Der Kommentar wird mit einer waagerechten Linie dargestellt

Der Standardwert für den Parameter ist "-".

## 2.5.4 MARK

Über den Befehl **mark** kann eine Markierung im Diagramm dargestellt werden. Eine Markierung ist eine zeitpunktbezogene Marke, die im Gegensatz zum **linecomment** am Diagrammrand erscheint.

### ➤ Syntax

**mark**: *instanceid*, *text*, *orientation*

### ➤ Parameter

- *instanceid*

Der Identifikator der Lebenslinie, mit der die Markierung verbunden werden soll.

- *text*

Ein beliebiger Text, der auf der Markierungslinie erscheinen soll. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.

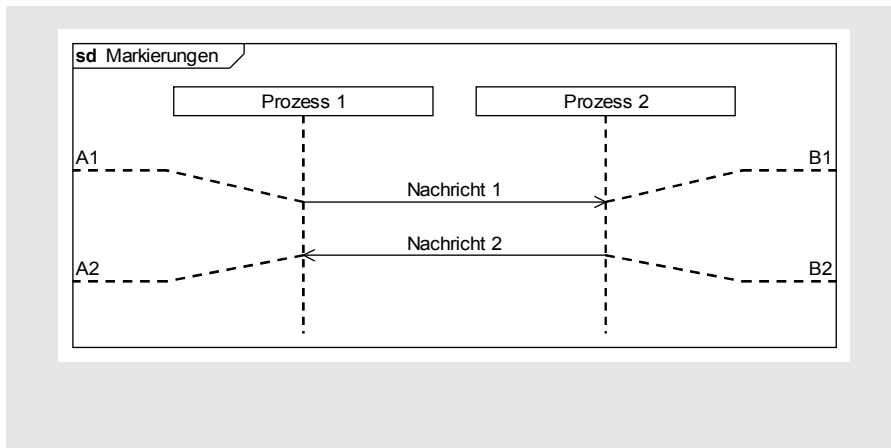
- *orientation* (optional)

Legt die Position der Markierung bezüglich fest.

Wert	Bedeutung
l	Die Markierung wird an der linken Seite des Diagramms dargestellt.
r	Die Markierung wird an der rechten Seite des Diagramms dargestellt.
t	Die Markierung wird oben herum geführt.
b	Die Markierung wird unten herum geführt.

Eine Kombination der Werte ist zulässig.  
Der Standardwert für den Parameter ist "lt".

➤ **Beispiel**



**DiagramName:** Markierungen  
**DiagramStyle:** uml  
**PageSize:** 550,auto  
**PageMargins:** 10,10,10,10  
**Left:** 50  
**Right:** 50  
  
**process:** p1, Prozess 1  
**process:** p2, Prozess 2  
;  
**msg:** p1, p2, Nachricht 1  
**mark:** p1, A1  
**mark:** p2, B1, r;  
**msg:** p2, p1, Nachricht 2  
**mark:** p1, A2, lb  
**mark:** p2, B2, rb;

## 2.6 Timer

Timerelemente werden in Sequenzdiagrammen zum Beschreiben zeitgenauer Abläufe verwendet. Sie Zeitangaben können sowohl als Durchlaufzeitdefinitionen als auch Zeitvorgaben dienen.

### 2.6.1 TIMERBEGIN

Über die Befehle **timerbegin** und **timerend** kann im Diagramm eine Zeitangabe modelliert werden. Dabei wird durch den Befehl **timerbegin** der Anfang der Zeitangabe definiert. Die Darstellung der Zeitangabe ist zwischen dem Startpunkt und dem Endpunkt mit einer Linie verbunden. Soll eine Darstellung ohne diese Linie erfolgen, so können hierfür die Befehle **settimer**, **stoptimer** und **timeout** verwendet werden.

➤ **Syntax**

**timerbegin:** timerid, instanceid, text, orientation, style

➤ **Parameter**

- **timerid**  
Der Identifikator des Timers. Der Identifikator wird für den Befehl **timerend** benötigt, damit dieser der richtigen Zeitangabe zugeordnet werden kann.
- **instanceid**  
Der Identifikator der Lebenslinie an der die Zeitangabe dargestellt werden soll.
- **text** (optional)  
Ein beliebiger Text, der am Anfang der Zeitangabe erscheinen soll. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.

- **orientation** (optional)

Legt die Position der Zeitangabe an der durch den Parameter **instanceid** definierten Lebenslinie fest.

Wert	Bedeutung
l	Die Zeitangabe wird links von der Lebenslinie dargestellt
r	Die Zeitangabe wird rechts von der Lebenslinie dargestellt

Der Standardwert für den Parameter ist "l".

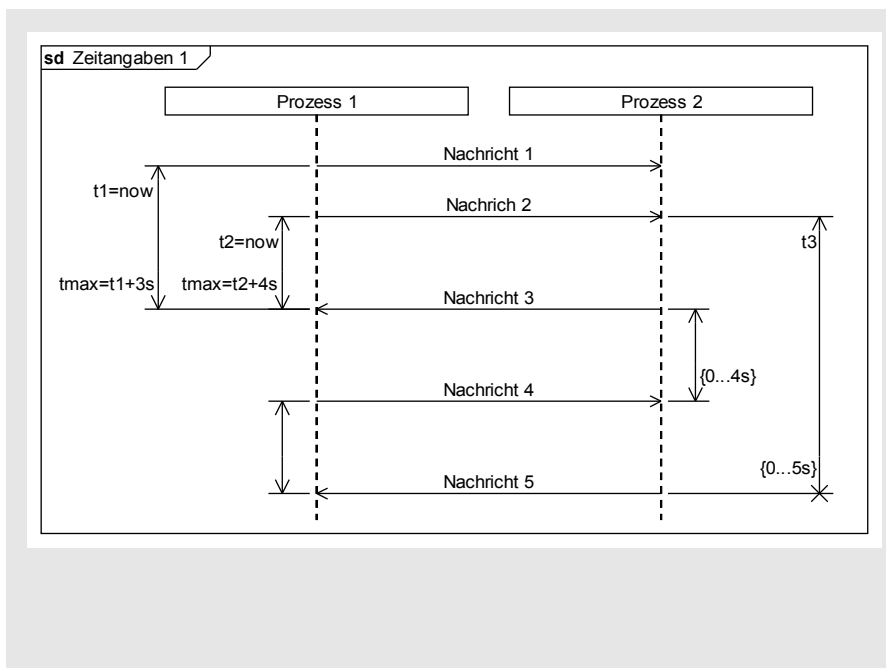
- **style** (optional)

Legt die Darstellungsart der Zeitangabe fest.

Wert	Bedeutung
n	Die Zeitangabe wird nah an der Lebenslinie gezeichnet.
o	Die Zeitangabe wird entfernt von der Lebenslinie gezeichnet, die Textposition ist außen.
i	Die Zeitangabe wird entfernt von der Lebenslinie gezeichnet, die Textposition ist innen.

Der Standardwert für den Parameter ist "n".

### ➤ Beispiel



**DiagramName:** Zeitangaben 1  
**DiagramStyle:** uml  
**PageSize:** 620, auto  
**PageMargins:** 10, 10, 10, 10

**process:** p1, "Prozess 1",,,70  
**process:** p2, "Prozess 2",,,0,20

**timerbegin:** t1, p1, t1=now,l,o  
**msg:** p1, p2, Nachricht 1;  
**timerbegin:** t2, p1, t2=now,l,n  
**timerbegin:** t3, p2,t3,r,i  
**msg:** p1, p2, Nachricht 2;  
;  
**timerbegin:** t4, p2,,r,n  
**msg:** p2, p1, Nachricht 3  
**timerend:** t1, tmax=t1+3s  
**timerend:** t2, tmax=t2+4s;  
;  
**timerbegin:** t5, p1  
**msg:** p1, p2, Nachricht 4  
**timerend:** t4,{0...4s};  
;  
**msg:** p2, p1, Nachricht 5  
**timerend:** t5  
**timerend:** t3, {0...5s},\*



## 2.6.2 TIMEREND

Über die Befehle **timerbegin** und **timerend** kann im Diagramm eine Zeitangabe modelliert werden. Dabei wird durch den Befehl **timerend** das Ende der Zeitangabe definiert. Die Darstellung der Zeitangabe ist zwischen dem Startpunkt und dem Endpunkt mit einer Linie verbunden. Soll eine Darstellung ohne diese Linie erfolgen, so können hierfür die Befehle **settimer**, **stoptimer** und **timeout** verwendet werden.

### ➤ Syntax

**timerend**: timerid, text, style

### ➤ Parameter

- timerid

Der Identifikator des Timers für die der Endpunkt definiert wird.

- text (optional)

Ein beliebiger Text, der am Ende der Zeitangebe erscheinen soll. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.

- style (optional)

Legt die Darstellung des Zeitangabeendpunkts fest.

Wert	Bedeutung
-	Die Zeitangabe wird ordnungsmäßig terminiert
*	Die Zeitangabe wird abgebrochen

Der Standardwert für den Parameter ist "-".

## 2.6.3 SETTIMER

Über den Befehl **settimer** kann im Diagramm der Anfang einer Zeitangabe modelliert werden. Die Darstellung dieser Zeitangabe ist zwischen dem Startpunkt und dem Endpunkt nicht mit einer Linie verbunden. Da der UML - Standard eine solche Darstellung von Zeitangeben nicht spezifiziert, wird im Diagramm die Darstellung des SDL - Standards verwendet. Um das Ende der Zeitangabe zu modellieren, muss der Befehl **stoptimer** bzw. **timeout** verwendet werden.

### ➤ Syntax

**settimer**: instanceid, text, orientation

➤ **Parameter**

- **instanceid**

Der Identifikator der Lebenslinie an der die Zeitangabe dargestellt werden soll.

- **text** (optional)

Ein beliebiger Text, der am Anfang der Zeitangabe erscheinen soll. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.

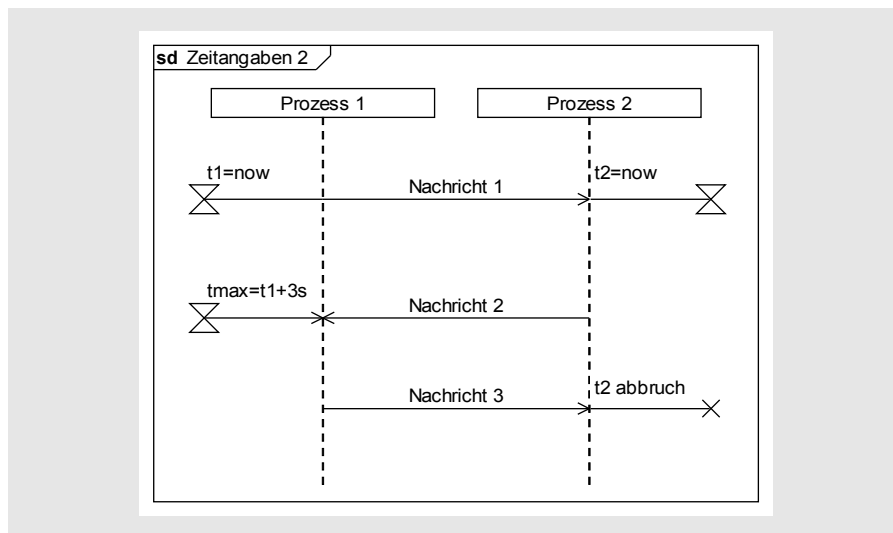
- **orientation** (optional)

Legt die Position der Zeitangabe an der durch den Parameter **instanceid** definierten Lebenslinie fest.

Wert	Bedeutung
l	Die Zeitangabe wird links von der Lebenslinie dargestellt
r	Die Zeitangabe wird rechts von der Lebenslinie dargestellt

Der Standardwert für den Parameter ist "l".

➤ **Beispiel**



**DiagramName:** Zeitangaben 2  
**DiagramStyle:** uml  
**PageSize:** auto, auto  
**PageMargins:** 10, 10, 10, 10

**process:** p1, "Prozess 1" ,,20  
**process:** p2, "Prozess 2" ,,0,20

**settimer:** p1, t1=now  
**settimer:** p2, t2=now,r  
**msg:** p1, p2, Nachricht 1;  
;  
**msg:** p2, p1, Nachricht 2  
**timeout:** p1, tmax=t1+3s;  
;  
**msg:** p1, p2, Nachricht 3  
**stoptimer:** p2,t2 abbruch,r;

## 2.6.4 STOPTIMER

Über den Befehl **stoptimer** kann im Diagramm der Abbruch einer Zeitangabe modelliert werden. Die Darstellung dieser Zeitangabe ist zwischen dem Startpunkt und dem Abbruchpunkt nicht mit einer Linie verbunden. Da der UML - Standard eine solche Darstellung von Zeitangeben nicht spezifiziert, wird im Diagramm die Darstellung des SDL - Standards verwendet. Um den Anfang der Zeitangabe zu modellieren, muss der Befehl **settimer** verwendet werden.

➤ **Syntax**

**stoptimer**: `instanceid`, `text`, `orientation`

➤ **Parameter**

- `instanceid`

Der Identifikator der Lebenslinie an der der Abbruch der Zeitangabe dargestellt werden soll.

- `text` (optional)

Ein beliebiger Text, der mit dem Abbruch der Zeitangabe erscheinen soll. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.

- `orientation` (optional)

Legt die Position des Abbruchs bezüglich der durch den Parameter `instanceid` definierten Lebenslinie fest.

Wert	Bedeutung
l	Der Abbruch der Zeitangabe wird links von der Lebenslinie dargestellt.
r	Der Abbruch der Zeitangabe wird rechts von der Lebenslinie dargestellt.

Der Standardwert für den Parameter ist "l".

## 2.6.5 TIMEOUT

Über den Befehl **timeout** kann im Diagramm das Ende einer Zeitangabe modelliert werden. Die Darstellung dieser Zeitangabe ist zwischen dem Startpunkt und dem Endpunkt nicht mit einer Linie verbunden. Da der UML - Standard eine solche Darstellung von Zeitangeben nicht spezifiziert, wird im Diagramm die Darstellung des SDL - Standards verwendet. Um den Anfang der Zeitangabe zu modellieren, muss der Befehl **settimer** verwendet werden.

➤ **Syntax**

**timeout**: `instanceid`, `text`, `orientation`

➤ **Parameter**

- `instanceid`

Der Identifikator der Lebenslinie an der das Ende der Zeitangabe dargestellt werden soll.

- **text** (optional)  
Ein beliebiger Text, der am Ende der Zeitangabe erscheinen soll. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.
- **orientation** (optional)  
Legt die Position des Endpunkts bezüglich der durch den Parameter **instanceid** definierten Lebenslinie fest.

Wert	Bedeutung
l	Der Endpunkt der Zeitangabe wird links von der Lebenslinie dargestellt.
r	Der Endpunkt der Zeitangabe wird rechts von der Lebenslinie dargestellt.

Der Standardwert für den Parameter ist "l".

## 2.7 Zeitmessungen

Diese SDL – spezifischen Diagrammelemente erlauben es Zeitmessungen in Sequenzdiagrammen zu spezifizieren und die Ergebnisse in Variablen zu Speichern. Diese Variablen können im weiteren Sequenzdiagrammverlauf verwendet werden um beispielsweise Intervalle zu definieren.

### 2.7.1 MEASUREBEGIN

Über die Befehle **measurebegin** und **measureend** kann im Diagramm eine Messung modelliert werden. Dabei wird durch den Befehl **measurebegin** der Anfang der Messung definiert. Die Darstellung der Messung ist zwischen dem Startpunkt und dem Endpunkt mit einer Linie verbunden. Eine Verschachtelung mehrerer Messungen an einer Lebenslinie ist nicht möglich. Verwenden sie hierfür die Befehle **measurestart** und **measurestop**, die ohne Verbindungslinie dargestellt werden.

#### ➤ Syntax

**measurebegin**: **instanceid**, **text**, **orientation**, **style**

#### ➤ Parameter

- **instanceid**

Der Identifikator der Lebenslinie an der die Messung dargestellt werden soll.

- **text** (optional)  
Ein beliebiger Text, der am Anfang der Messung erscheinen soll. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.
- **orientation** (optional)  
Legt die Position der Messung an der durch den Parameter **instanceid** definierten Lebenslinie fest.

Wert	Bedeutung
l	Die Messung wird links von der Lebenslinie dargestellt
r	Die Messung wird rechts von der Lebenslinie dargestellt

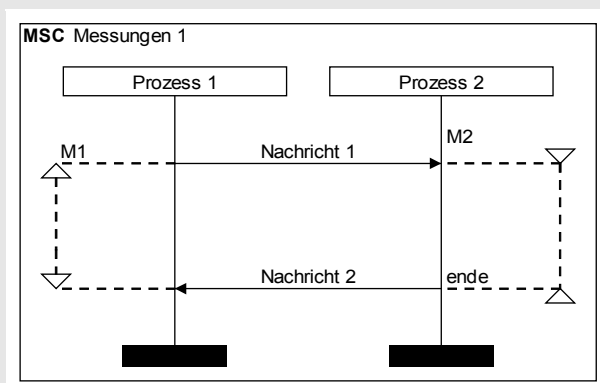
Der Standardwert für den Parameter ist "l".

- **style** (optional)  
Legt die Darstellungsart der Messung fest.

Wert	Bedeutung
-	Die Messpfeile werden unterhalb der horizontalen Messlinie dargestellt.
*	Die Messpfeile werden oberhalb der horizontalen Messlinie dargestellt.

Der Standardwert für den Parameter ist "-".

### ➤ Beispiel



**DiagramName:** Messungen 1  
**DiagramStyle:** sdl  
**PageSize:** auto, auto  
**PageMargins:** 10, 10, 10, 10  
**left:**10  
**right:** 10  
**process:** p1, "Prozess 1"  
**process:** p2, "Prozess 2"  
**measurebegin:** p1, M1  
**measurebegin:** p2, M2, r, \*  
**msg:** p1, p2, Nachricht 1;  
;  
;  
**msg:** p2, p1, Nachricht 2  
**measureend:** p1  
**measureend:** p2, ende;

## 2.7.2 MEASUREEND

Über die Befehle **measurebegin** und **measureend** kann im Diagramm eine Messung modelliert werden. Dabei wird durch den Befehl **measureend** das Ende der Messung definiert. Die Darstellung der Messung ist zwischen dem Startpunkt und dem Endpunkt mit einer Linie verbunden. Eine Verschachtelung mehrerer Messungen an einer Lebenslinie ist nicht möglich. Verwenden sie hierfür die Befehle **measurestart** und **measurestop**, die ohne Verbindungslinie dargestellt werden.

### ➤ Syntax

**measureend**: `instanceid`, `text`

### ➤ Parameter

- `instanceid`

Der Identifikator der Lebenslinie an der die Messung beendet werden soll.

- `text` (optional)

Ein beliebiger Text, der am Ende der Messung erscheinen soll. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.

## 2.7.3 MEASURESTART

Über den Befehl **measurestart** kann im Diagramm der Anfang einer Messung modelliert werden. Die Darstellung dieser Messung ist zwischen dem Startpunkt und dem Endpunkt nicht mit einer Linie verbunden. Um das Ende der Zeitangabe zu modellieren, muss der Befehl **measurestop** verwendet werden.

### ➤ Syntax

**measurestart**: `instanceid`, `gate`, `text`, `orientation`, `style`

### ➤ Parameter

- `instanceid`

Der Identifikator der Lebenslinie an der der Messanfang dargestellt werden soll.

- `gate` (optional)

Ein beliebiger Text, der am Gate der Messung erscheinen soll. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.

- **text** (optional)  
Ein beliebiger Text, der am Anfang der Messung erscheinen soll. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.
- **orientation** (optional)  
Legt die Position der Messung bezüglich der durch den Parameter **instanceid** definierten Lebenslinie fest.

Wert	Bedeutung
l	Die Messung wird links von der Lebenslinie dargestellt
r	Die Messung wird rechts von der Lebenslinie dargestellt

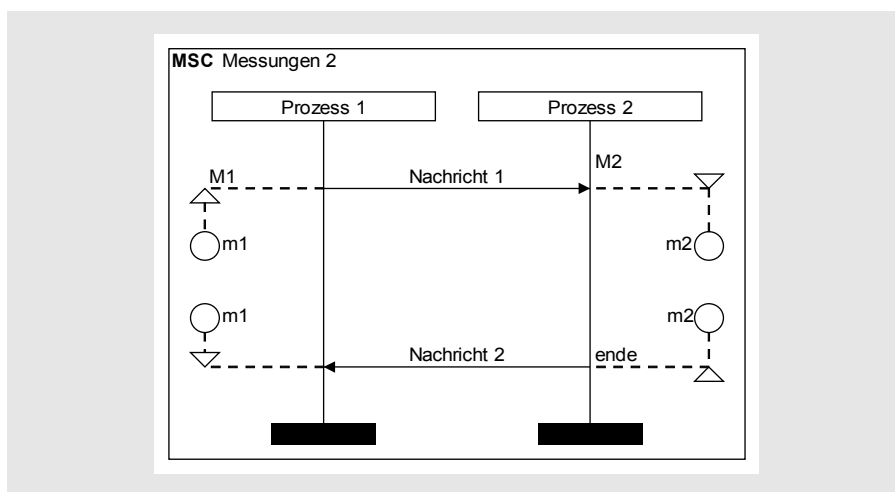
Der Standardwert für den Parameter ist "l".

- **style** (optional)  
Legt die Darstellungsart der Messung fest.

Wert	Bedeutung
-	Die Messpfeile werden unterhalb der horizontalen Messlinie dargestellt.
*	Die Messpfeile werden oberhalb der horizontalen Messlinie dargestellt.

Der Standardwert für den Parameter ist "-".

### ➤ Beispiel



**DiagramName:** Messungen 2  
**DiagramStyle:** sdl  
**PageSize:** auto, auto  
**PageMargins:** 10, 10, 10, 10  
**left:**10  
**right:** 10

**process:** p1, "Prozess 1"  
**process:** p2, "Prozess 2"  
  
**measurestart:** p1, m1,M1  
**measurestart:** p2, m2, M2,r,\*  
**msg:** p1, p2, Nachricht 1;  
;  
;  
;  
**msg:** p2, p1, Nachricht 2  
**measurestop:** p1,m1  
**measurestop:** p2, m2, ende,r,\*;

## 2.7.4 MEASURESTOP

Über den Befehl **measurestop** kann im Diagramm das Ende einer mit dem Befehl **measurestart** definierten Messung, modelliert werden. Die Darstellung dieser Messung ist zwischen dem Startpunkt und dem Endpunkt nicht mit einer Linie verbunden.

### ➤ Syntax

**measurestop**: *instanceid*, *gate*, *text*, *orientation*, *style*

### ➤ Parameter

- **instanceid**  
Der Identifikator der Lebenslinie an der das Messende dargestellt werden soll.
- **gate** (optional)  
Ein beliebiger Text, der am Gate der Messung erscheinen soll. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.
- **text** (optional)  
Ein beliebiger Text, der am Ende der Messung erscheinen soll. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.
- **orientation** (optional)  
Legt die Position des Messendes bezüglich der durch den Parameter **instanceid** definierten Lebenslinie fest.

Wert	Bedeutung
l	Das Messende wird links von der Lebenslinie dargestellt.
r	Das Messende wird rechts von der Lebenslinie dargestellt.

Der Standardwert für den Parameter ist "l".

- **style** (optional)  
Legt die Darstellungsart der Messung fest.

Wert	Bedeutung
-	Der Messpfeil wird unterhalb der horizontalen Messlinie dargestellt.
*	Der Messpfeil wird oberhalb der horizontalen Messlinie dargestellt.



Der Standardwert für den Parameter ist "-".

## 2.8 Kombinierte Fragmente

Kombiniertes Fragment ist eine allgemeine Bezeichnung für eine abgeschlossene Interaktionseinheit innerhalb des Sequenzdiagramms. Dies können beispielsweise Alternativen, Parallelitäten oder Schleifen sein.

### 2.8.1 FRAGMENTBEGIN

Über die Befehle **fragmentbegin** und **fragmentend** kann im Diagramm ein kombiniertes Fragment (engl. CombinedFragment) modelliert werden. Dabei wird durch den Befehl **fragmentbegin** der Anfang des Fragments definiert. Ein kombiniertes Fragment ist eine allgemeine Bezeichnung für folgende abgeschlossene Interaktionseinheiten: Abbruch, Alternative, irrelevante Nachrichten, kritischer Bereich, Negation, Option, Parallelität, Schleife, schwache Sequenz, Sicherstellung und strikte Sequenz.

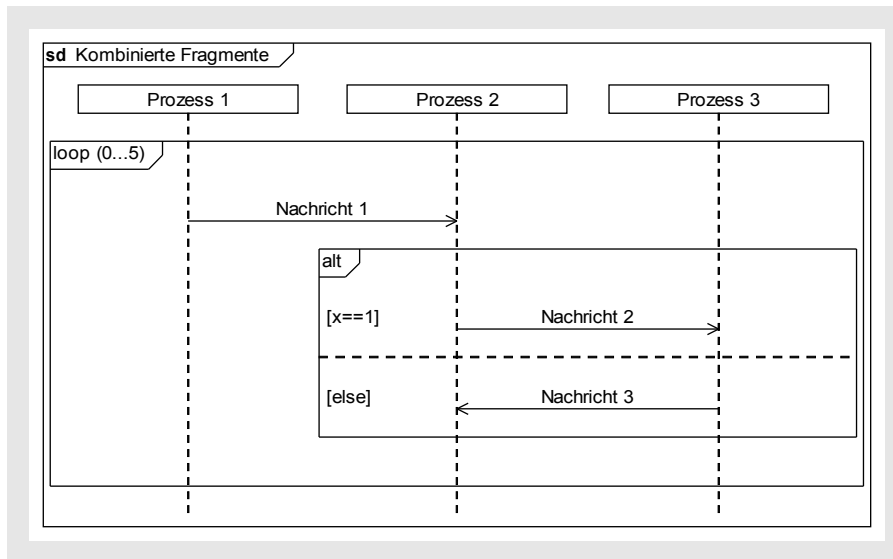
#### ➤ Syntax

**fragmentbegin**: *fragmentid*, *firstinstanceid*, *lastinstanceid*, *text*

#### ➤ Parameter

- **fragmentid**  
Der Identifikator des kombinierten Fragments.
- **firstinstanceid**  
Der Identifikator der ersten Lebenslinie, über die sich das kombinierte Fragment erstrecken soll.
- **lastinstanceid**  
Der Identifikator der letzten Lebenslinie, über die sich das kombinierte Fragment erstrecken soll.
- **text**  
Ein beliebiger Text, der die Interaktion des kombinierten Fragments definiert. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.

➤ **Beispiel**



**DiagramName:** Kombinierte Fragmente  
**DiagramStyle:** uml  
**PageSize:** auto, auto  
**PageMargins:** 10, 10, 10, 10

**process:** p1, "Prozess 1"  
**process:** p2, "Prozess 2"  
**process:** p3, "Prozess 3"

**fragmentbegin:** i1, p1,p3, loop (0...5);  
**msg:** p1, p2, Nachricht 1;  
**fragmentbegin:** i2, p2,p3,alt;  
**fragmenttext:** i2, "[x==1]"  
**msg:** p2, p3, Nachricht 2;  
**fragmentseparator:** i2;  
**fragmenttext:** i2, [else]  
**msg:** p3, p2, Nachricht 3;  
**fragmentend:** i2;  
**fragmentend:** i1

## 2.8.2 FRAGMENTEND

Über die Befehle **fragmentbegin** und **fragmentend** kann im Diagramm ein kombiniertes Fragment (engl. CombinedFragment) modelliert werden. Dabei wird durch den Befehl **fragmentend** das Ende des Fragments definiert. Ein kombiniertes Fragment ist eine allgemeine Bezeichnung für folgende abgeschlossene Interaktionseinheiten: Abbruch, Alternative, irrelevante Nachrichten, kritischer Bereich, Negation, Option, Parallelität, Schleife, schwache Sequenz, Sicherstellung und strikte Sequenz.

➤ **Syntax**

**fragmentend:** fragmentid

➤ **Parameter**

- **fragmentid**

Der Identifikator des kombinierten Fragments, das beendet werden soll.

## 2.8.3 FRAGMENTSEPARATOR

Über die Befehle **fragmentseparator** kann in einem kombinierten Fragment eine Trennlinie modelliert werden.

➤ **Syntax**

**fragmentseparator:** fragmentid

➤ **Parameter**

- **fragmentid**

Der Identifikator des kombinierten Fragments, für das die Trennlinie modelliert werden soll.

## 2.8.4 FRAGMENTTEXT

Über die Befehle **fragmenttext** kann in einem kombinierten Fragment ein Text innerhalb des Fragments definiert werden. Der Text wird an der linken Seite dargestellt.

➤ **Syntax**

**fragmenttext:** fragmentid, text

➤ **Parameter**

- **fragmentid**

Der Identifikator des kombinierten Fragments, in dem der Text erscheinen soll.

- **text**

Ein beliebiger Text, der in dem kombinierten Fragment erscheinen soll. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.

## 2.9 REFERENZEN

Die Referenzen innerhalb eines Sequenzdiagramms erlauben es, auf weitere Sequenzdiagramme die eine Aktion ausführlich beschreiben zu verweisen.

### 2.9.1 REF

Mit dem Befehl **ref** kann im Diagramm eine Interaktionsreferenz modelliert werden.

➤ **Syntax**

**ref:** firstinstanceid, lastinstanceid, text

➤ **Parameter**

- **firstinstanceid**

Der Identifikator der ersten Lebenslinie, über die sich die Interaktionsreferenz erstrecken soll.

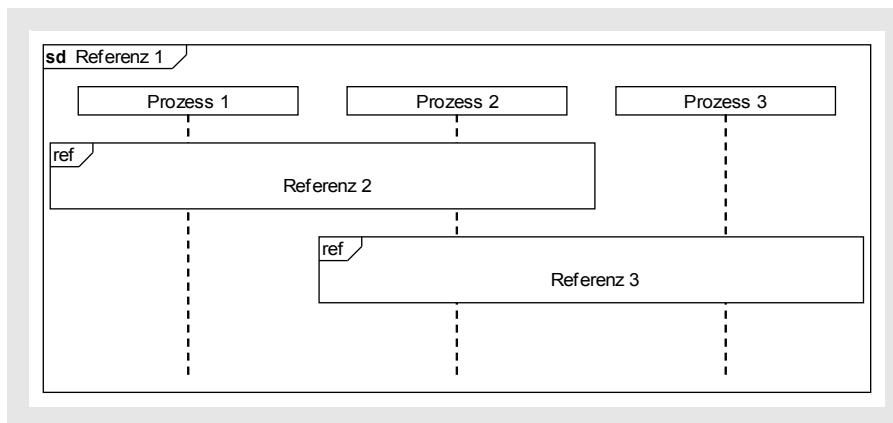
- **lastinstanceid**

Der Identifikator der letzten Lebenslinie, über die sich die Interaktionsreferenz erstrecken soll.

- **text**

Ein beliebiger Text, der den Namen und ggf. die Parameter des referenzierten Interaktionsdiagramms definiert. Sollen im Text Satzzeichen verwendet werden, muss der Text in Anführungszeichen angegeben werden.

### ➤ Beispiel



**DiagramName:** Referenz 1  
**DiagramStyle:** uml  
**PageSize:** auto, auto  
**PageMargins:** 10, 10, 10, 10

**process:** p1, "Prozess 1"  
**process:** p2, "Prozess 2"  
**process:** p3, "Prozess 3"

**ref:** p1,p2, "Referenz 2";  
**ref:** p2,p3, "Referenz 3";

## 3 Steuerzeichen

Steuerzeichen innerhalb der Diagrammbeschreibung spielen eine große Rolle bei der Generierung der Sequenzdiagramme. Sie sind unter anderem für den vertikalen Versatz von Diagrammelementen und für einen manuellen Zeilenumbruch innerhalb von Diagrammtexten verantwortlich. Dieses Kapitel beschreibt die verfügbaren Steuerzeichen und ihre Funktion.

### 3.1 Textformatierung

Die Steuerzeichen in Diagrammtexten sind für einen manuellen Zeilenumbruch und für die Darstellung von Satzzeichen zuständig.

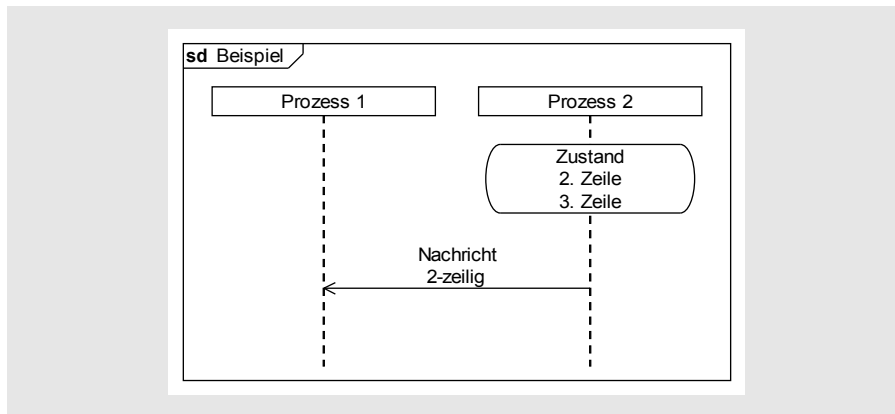
#### 3.1.1 \n – Zeilenumbruch

Über das Steuerzeichen "**\n**" kann in Parametern vom Typ **text** ein Zeilenumbruch erzwungen werden.

#### ➤ Syntaxbeispiel

**befehl:** parameter1, parameter\n2

➤ **Beispiel**



**DiagramName:** Beispiel  
**DiagramStyle:** uml  
**PageSize:** 400, auto  
**PageMargins:** 10, 10, 10, 10

**process:** p1, "Prozess 1"  
**process:** p2, "Prozess 2"  
**state:** p2, Zustand\n2. Zeile\n3. Zeile;  
**msg:** p2, p1, Nachricht\n2-zeilig;

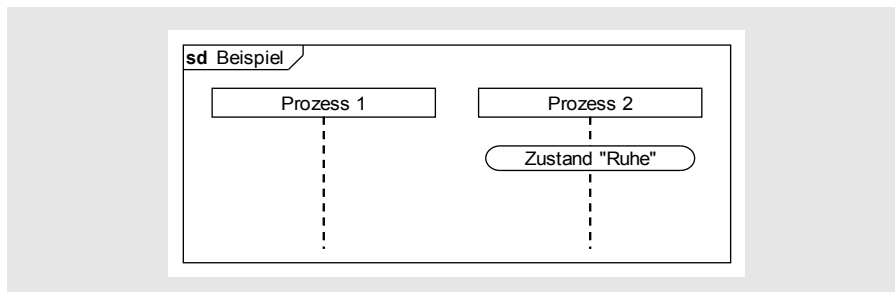
### 3.1.2 \" – Anführungszeichen

Über das Steuerzeichen "\" kann in Parametern vom Typ `text` ein Anführungszeichen dargestellt werden.

➤ **Syntaxbeispiel**

**befehl:** parameter1, \"parameter 2\"

➤ **Beispiel**



**DiagramName:** Beispiel  
**DiagramStyle:** uml  
**PageSize:** 400, auto  
**PageMargins:** 10, 10, 10, 10

**process:** p1, "Prozess 1"  
**process:** p2, "Prozess 2"  
**state:** p2, Zustand \"Ruhe\";

## 3.2 Diagrammformatierung

Die Steuerzeichen der Diagrammformatierung sind für den vertikalen Versatz der Diagrammelemente und einen manuellen Seitenumbruch zuständig

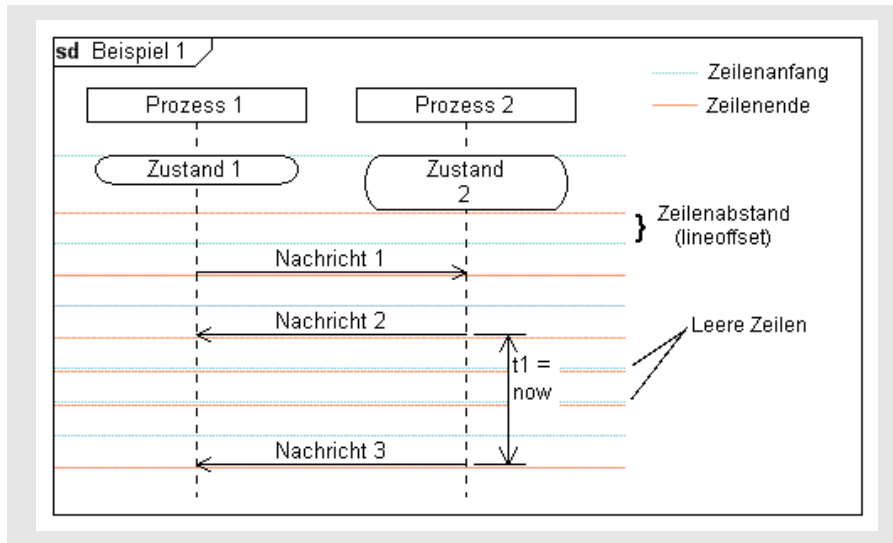
### 3.2.1 ; – Diagrammzeilenumbruch

Über das Steuerzeichen Semikolon ";" am Ende einer Textzeile wird das Ende der Diagrammzeile definiert. Alle Diagrammelemente, die sich in einer Diagrammzeile befinden, werden Vertikal auf gleicher Höhe dargestellt.

➤ **Syntaxbeispiel**

**befehl:** parameter1, parameter2;

➤ **Beispiel**



**DiagramName:** Beispiel 1  
**DiagramStyle:** uml  
**PageSize:** 500, auto  
**PageMargins:** 10, 10, 10, 10  
**right:** 150

```

process: p1, "Prozess 1"
process: p2, "Prozess 2"

state: p1, Zustand 1
state: p2, Zustand\n2;
msg: p1, p2, Nachricht 1;

msg: p2, p1, Nachricht 2
timerbegin: t1, p2, t1 = now,r;
;
;
msg: p2, p1, Nachricht 3
timerend: t1
    
```

### 3.2.2 { } – kein Diagrammzeilenumbruch

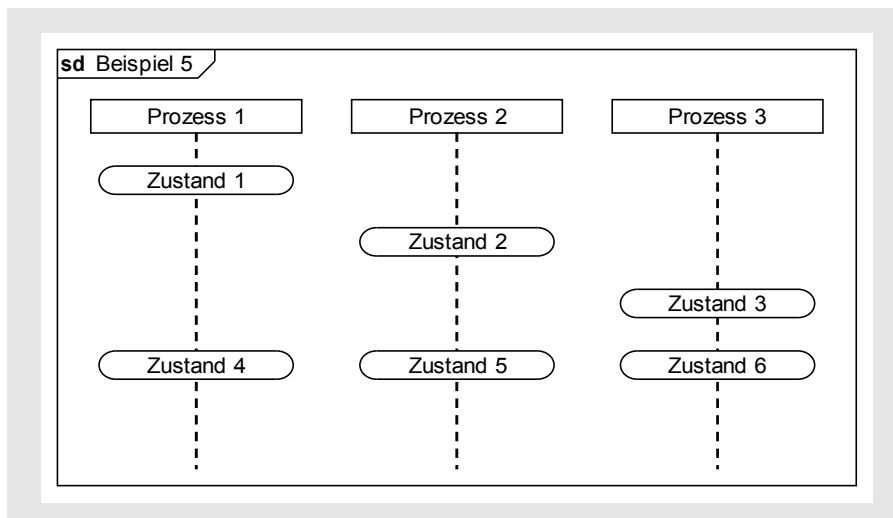
Über die Verwendung geschweifter Klammern "{" und "}" kann der Diagrammzeilenumbruch durch das Semikolon unterdrückt werden.

➤ **Syntaxbeispiel**

```

{
befehl1: parameter1, parameter2;
befehl2: parameter1, parameter2;
}
    
```

➤ **Beispiel**



**DiagramName:** Beispiel 5  
**DiagramStyle:** uml  
**PageSize:** 500, auto  
**PageMargins:** 10, 10, 10, 10

```

process: p1, "Prozess 1"
process: p2, "Prozess 2"
process: p3, "Prozess 3"

state: p1, Zustand 1;
state: p2, Zustand 2;
state: p3, Zustand 3;

{
state: p1, Zustand 4;
state: p2, Zustand 5;
state: p3, Zustand 6;
}
    
```

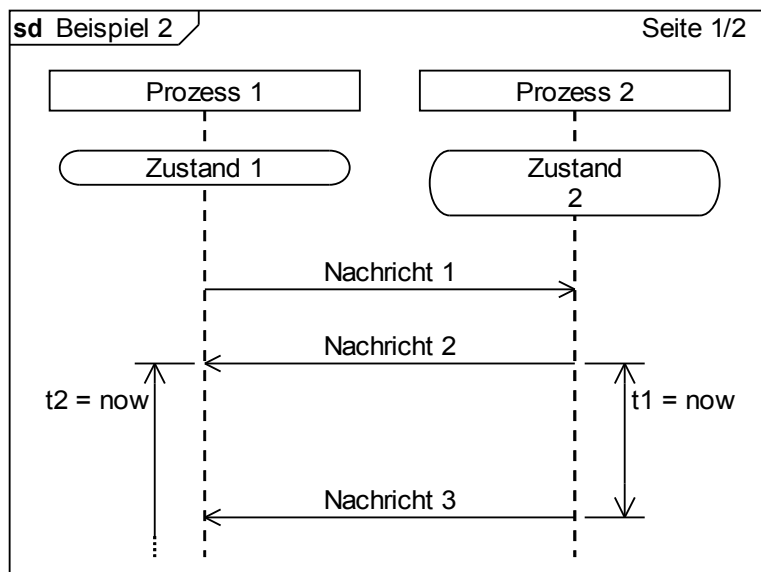
### 3.2.3 ;; – manueller Seitenumbruch

Über das doppelte Steuerzeichen Semikolon ";;" am Ende einer Textzeile wird ein manueller Seitenumbruch definiert.

#### ➤ Syntaxbeispiel

**befehl:** `parameter1, parameter2;;`

#### ➤ Beispiel



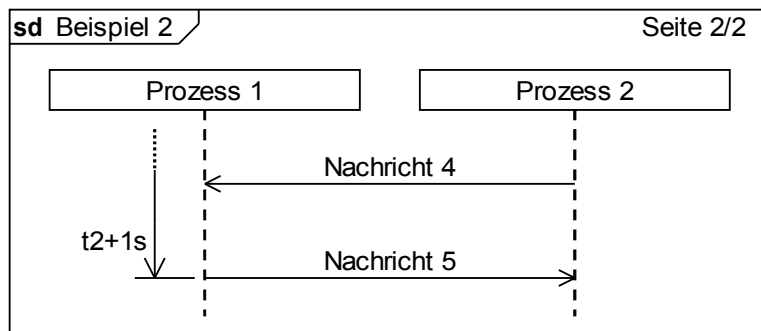
**DiagramName:** Beispiel 2  
**DiagramStyle:** uml  
**PageSize:** 400, auto  
**PageMargins:** 10, 10, 10, 10

**process:** p1, "Prozess 1"  
**process:** p2, "Prozess 2"

**state:** p1, Zustand 1  
**state:** p2, Zustand 2;  
**msg:** p1, p2, Nachricht 1;















**msg:** p2, p1, Nachricht 2  
**timerbegin:** t2, p1, t2 = now  
**timerbegin:** t1, p2, t1 = now,r;  
;  
;  
**msg:** p2, p1, Nachricht 3  
**timerend:** t1;;





































**msg:** p2, p1, Nachricht 4;  
**msg:** p1, p2, Nachricht 5  
**timerend:** t2, t2+1s





































## Anhang A


### Farbennamen

	aliceblue	#f0f8ff
	antiquewhite	#faebd7
	aqua	#00ffff
	aquamarine	#7fffd4
	azure	#f0ffff
	beige	#f5f5dc
	bisque	#ffe4c4
	blanchedalmond	#ffebcd
	blue	#0000ff
	blueviolet	#8a2be2
	brown	#a52a2a
	burlywood	#deb887
	cadetblue	#5f9ea0
	chartreuse	#7fff00
	chocolate	#d2691e
	coral	#ff7f50
	cornflowerblue	#6495ed
	cornsilk	#fff8dc
	crimson	#dc143c
	cyan	#00ffff
	darkblue	#00008b
	darkcyan	#008b8b
	darkgoldenrod	#b8860b
	darkgreen	#006400
	darkkhaki	#bdb76b
	darkmagenta	#8b008b
	darkolivegreen	#556b2f
	darkorange	#ff8c00
	darkorchid	#9932cc
	darkred	#8b0000
	darksalmon	#e9967a
	darkseagreen	#8fbc8f
	darkslateblue	#483d8b
	darkslategray	#2f4f4f
	darkturquoise	#00ced1
	darkviolet	#9400d3

	deeppink	#ff1493
	deepskyblue	#00bfff
	dodgerblue	#1e90ff
	firebrick	#b22222
	floralwhite	#fffaf0
	forestgreen	#228b22
	fuchsia	#ff00ff
	ghostwhite	#f8f8ff
	gold	#ffd700
	goldenrod	#daa520
	green	#008000
	greenyellow	#adff2f
	honeydew	#f0ffff
	hotpink	#ff69b4
	indianred	#cd5c5c
	indigo	#4b0082
	ivory	#fffff0
	khaki	#f0e68c
	lavender	#e6e6fa
	lavenderblush	#fff0f5
	lawngreen	#7cfc00
	lemonchiffon	#fffacd
	lightblue	#add8e6
	lightcoral	#f08080
	lightcyan	#e0ffff
	lightgoldenrodyellow	#fafad2
	lightgreen	#90ee90
	lightpink	#ffb6c1
	lightsalmon	#ffa07a
	lightseagreen	#20b2aa
	lightskyblue	#87cefa
	lightslategray	#778899
	lightsteelblue	#b0c4de
	lightyellow	#ffffe0
	lime	#00ff00
	limegreen	#32cd32



	linen	#faf0e6
	magenta	#ff00ff
	maroon	#800000
	mediumaquamarine	#66cdaa
	mediumblue	#0000cd
	mediumorchid	#ba55d3
	mediumpurple	#9370db
	mediumseagreen	#3cb371
	mediumslateblue	#7b68ee
	mediumspringgreen	#00fa9a
	mediumturquoise	#48d1cc
	mediumvioletred	#c71585
	midnightblue	#191970
	mintcream	#f5fffa
	mistyrose	#ffe4e1
	moccasin	#ffe4b5
	navajowhite	#ffdead
	navy	#000080
	oldlace	#fdf5e6
	olive	#808000
	olivedrab	#6b8e23
	orange	#ffa500
	orangered	#ff4500
	orchid	#da70d6
	palegoldenrod	#eee8aa
	palegreen	#98fb98
	paleturquoise	#afeeee
	palevioletred	#db7093
	papayawhip	#ffefd5
	peachpuff	#ffdab9
	peru	#cd853f
	pink	#ffc0cb
	plum	#dda0dd
	powderblue	#b0e0e6

	purple	#800080
	red	#ff0000
	rosybrown	#bc8f8f
	royalblue	#4169e1
	saddlebrown	#8b4513
	salmon	#fa8072
	sandybrown	#f4a460
	seagreen	#2e8b57
	seashell	#fff5ee
	sienna	#a0522d
	skyblue	#87ceeb
	slateblue	#6a5acd
	slategray	#708090
	snow	#fffafa
	springgreen	#00ff7f
	steelblue	#4682b4
	tan	#d2b48c
	teal	#008080
	thistle	#d8bfd8
	tomato	#ff6347
	turquoise	#40e0d0
	violet	#ee82ee
	wheat	#f5deb3
	whitesmoke	#f5f5f5
	yellow	#ffff00
	yellowgreen	#9acd32
	white	#ffffff
	gainsboro	#dcdcdc
	lightgrey	#d3d3d3
	silver	#c0c0c0
	darkgray	#a9a9a9
	gray	#808080
	dimgray	#696969
	black	#000000

## 4 Anhang B

### *Index*

actor	21	msgbegin	28
author	14	msgend	30
backcolor	19	nextpage	13
comment	35	pagemargins	7
commentoverall	36	pagesize	4
company	15	printauthor	14
create	24	printcompany	15
date	16	printcreationdate	17
diagramname	8	printdate	17
diagramstyle	8	printfilename	16
dummyprocess	23	printfootline	13
fillcolor	19	printversion	18
font	9	process	22
found	30	ref	51
fragmentbegin	49	regionbegin	25
fragmentend	50	regionend	26
fragmentseparator	50	right	11
fragmenttext	51	settimer	41
left	10	state	32
linecomment	37	stateoverall	33
lineoffset	11	stop	24
lost	31	stoptimer	42
mark	38	task	34
measurebegin	49	textcolor	20
measureend	46	timeout	43
measurestart	46	timerbegin	39
measurestop	48	timerend	41
msg	27	version	18

ITESYS Institut für technische Systeme GmbH

Emil-Figge-Str. 76  
44227 Dortmund

Telefon: (0231) 97 42 71 10  
Telefax: (0231) 18 99 87 88

Internet: [www.itesys-gmbh.de](http://www.itesys-gmbh.de)  
E-mail: [info@itesys.de](mailto:info@itesys.de)