

**aspect Terminatable depends on Pausable**

*structural view*

!TerminatableParticipant

**TerminatableContext**

~ TerminatableContext create()  
+ pauseContext()  
+ continueContext()  
+ terminateContext(!TerminatableParticipant)

**!TerminatableParticipant**

+ leaveContext()

*Pausable instantiation*

**!PausableParticipant** →  
**!TerminatableParticipant**

*Pausable binding*

**TerminatableContext** →  
**PausableContext**  
leaveContext → leaveContext

*message view terminateContext* affected by Pausable.getParticipants, Pausable.contextCompleted

**Pointcut**

**caller: Caller**

**target:**  
**TerminatableContext**

terminateContext(terminator)

**Advice**

**caller: Caller**

**target:**  
**TerminatableContext**

terminateContext(terminator)

pauseContext()

participants := getParticipants()

loop [p in participants]

opt [p <> terminator]

**p:**  
**!TerminatableParticipant**

leaveContext()

**terminator:**  
**!TerminatableParticipant**

leaveContext()

contextCompleted()

*Default  
Instantiation*

**caller** → \*

**Caller** → \*

**target** → \*