

INestedParticipant

INestedParticipant
+ enterContext() + leaveContext()

Context instantiation
IParticipant →
INestedParticipant

NestedContext	
+ INestedContext getParent()	
~ setParent(INestedContext p)	
~ addChild(INestedContext c)	
+ Set<INestedContext> getChildren()	
	parent ◆ 0..1

children
0..*

Advice

caller: Caller

caller: Caller

target:
INestedParticipant

```
enterContext()
```

```
enterContext()
```

target:
edParticipant

```
oldContext :=
getContext()
```

oldContext:
NestedContext

```
opt [oldContext  $\diamond$  null]
┌
│   newContext :=
│   getContext()
└
```

newContext:
INestedContext

opt [newContext \diamond null]	
	addChild(newContext)
	setParent(olderContext)

<u>Default</u>
<u>Instantiation</u>
caller \rightarrow *
Caller \rightarrow *
target \rightarrow *

Advice

caller: Caller

caller: Caller

target:
INestedParticipant

```
leaveContext()
```

leaveContext()

target:
edParticipant

```
oldContext :=
getContext()
```

```
parentContext :=
getParent()
```

```
opt [parentContext < null] /
└─> setContext(parentContext)
```

<u>Default</u>
<u>Instantiation</u>
caller \rightarrow *
Caller \rightarrow *
target \rightarrow *

Advice

caller: Caller

caller: Caller

target:
NestedContext

```
result := m()
```

```
result := m()
```

target:
stedContext

```

result:
Collection<T>

```

recursive

```
children := getChildren()
```

```
loop [c within children]
```

```
childResult := m()
```

```
addAll(childResult)
```

c: NestedContext

<u>Default</u>
<u>Instantiation</u>
caller → *
Caller → *
target → *