

[Scala 3 Reference](#) / [Other Changed Features](#) / [Vararg Splices](#)**LEARN**

INSTALL

PLAYGROUND

FIND A LIBRARY

COMMUNITY

BLOG

# Vararg Splices

[Edit this page on GitHub](#)

The syntax of vararg splices in patterns and function arguments has changed. The new syntax uses a postfix `*`, analogously to how a vararg parameter is declared.

```

val arr = Array(0, 1, 2, 3)
val lst = List(arr*)           // vararg splice argument
lst match
  case List(0, 1, xs*) => println(xs) // binds xs to Seq(2, 3)
  case List(1, _) =>           // wildcard pattern

```

The old syntax for splice arguments will be phased out.

```

/*!*/ val lst = List(arr: _*) // syntax error
lst match
  case List(0, 1, xs @ _*) // ok, equivalent to `xs*`

```

## Syntax

```

ArgumentPatterns ::= '(' [Patterns] ')'
                  | '(' [Patterns ',' Pattern2] '*' ')'

ParArgumentExprs ::= '(' ['using'] ExprsInParens ')'
                  | '(' [ExprsInParens ',' PostfixExpr] '*' ')'

```


## Compatibility considerations

To enable cross compilation between Scala 2 and Scala 3, the compiler will accept both the old and the new syntax. Under the `-source future` setting, an error will be emitted when the old syntax is encountered. An automatic rewrite from old to new syntax is offered under `-source future-migration`.




▼ Match...


Full Screen

🔍

 Scaladoc

Copyright (c) 2002-2022, LAMP/EPFL



↑