

# Re: Nested &REST parameter vs. single named parameter

---

*Source:* <http://coding.derkeiler.com/Archive/Lisp/comp.lang.lisp/2005-11/msg00858.html>

---

- *From:* "nepheles" <nepheles@xxxxxxxxxxxxxx>
  - *Date:* 13 Nov 2005 10:09:31 -0800
- 

Michael J. Forster wrote:

> Hi,  
> ...  
> In another thread [2], Steven E. Harris suggested that this might be  
> just a matter of style, but that he found the choice of idiom  
> difficult. In that same thread, Barry Margolin responded:  
>  
> It's just a convenience feature that allows for terser  
> function calls. There's no semantic difference between:  
>  
> (defun fun1 (&rest args) ...)  
> (fun1 ...)  
>  
> and  
>  
> (defun fun2 (args) ...)  
> (fun2 (list ...))

Except in the case that fun1 destructively modifies args, and is running in a Lisp implementation that doesn't provide freshly-consed lists in &rest parameters (Zetalisp springs to mind).

.

---

- *References:*

- ◆ [\*Nested &REST parameter vs. single named parameter\*](#)  
◇ *From:* Michael J. Forster

- Prev by Date: [\*Re: Nested &REST parameter vs. single named parameter\*](#)
- Next by Date: [\*Re: Nested &REST parameter vs. single named parameter\*](#)
- Previous by thread: [\*Re: Nested &REST parameter vs. single named parameter\*](#)
- Next by thread: [\*Re: crosswords\*](#)
- Index(es):
  - ◆ [\*Date\*](#)
  - ◆ [\*Thread\*](#)