

Your load balancer wants to take a level of fighter and wizard



Lori MacVittie, 2009-05-05



If you've ever played [Dungeons & Dragons](#) for an extended period of time (a campaign, in the vernacular) you know that of all the classes available the cleric is the least likely to be chosen willingly. The cleric class is much like the kid picked last in kickball, chosen only because you *have* to, not because you want to. Okay, *bard* may actually be less likely but cleric is really, really close and you *need* a cleric, you don't necessarily *need* a bard.

The problem is that clerics can be somewhat dull to play but they are also an essential part of the game. Without them, well, it can be difficult if not impossible to succeed. They tend to the wounds that others received

while gloriously fighting off vile creatures. Clerics never get the glory because they aren't supposed to be on the front line fighting the evil, they're just supposed to stand back and *heal us for crying out loud*, before we fall down and have to roll up new characters. And hurry up about it, I only have like two hit points left!

Very few adventurous types want to play the cleric. Somewhere along the evolution of game design the folks behind [Dungeons & Dragons](#) recognized this fact and decided to make a concept called "multi-classing" much easier than it had been, which in turn would make choosing to play a cleric – if only part-time – a bit more palatable.

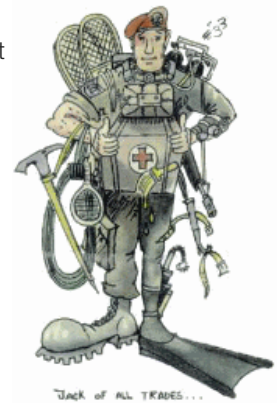
Multi-classing gives a character the ability to be more than one class at the same time. So rather than *just* be a cleric, they can also be a fighter or a ranger or a wizard or whatever they'd like to be. This means that the cleric could not only heal, but could fight, too, and when facing a dragon it really is important that you have as many people hacking away as possible. Really, trust me on this: it's important.

Multi-classing provides individual characters with a way to be more helpful to the entire group; it provides a broader set of capabilities in the face of limited resources. Kind of like application delivery...

MULTI-CLASSING YOUR LOAD BALANCER

You've probably got a [load-balancer](#) sitting around, providing value to be sure, but it's not necessarily very exciting. It only gets to add value when deciding where to load balance a request. Much like the cleric, [the load balancer is kind of bored](#) and really would like to get more involved and add more value to the group (the [applications](#)).

Some load balancers (ones that are really [application delivery controllers](#) but aren't being leveraged up to their full potential) can take a level of [acceleration](#), or optimization, or [security](#). They can "multi-class" into the different classes (functions) that make up application delivery, which ultimately adds a lot more value to the group and has the benefit of making the load balancer feel a lot more useful. It also means that you can get more functionality out of one person and *don't* have to add more adventurers (point solutions) among whom you will need to [spread the loot](#) (your budget). And if there's one thing that gamers and IT folks share, it's that they both operate in loot-constrained environments and want to share it as little as possible.



[Unified application delivery platforms](#) afford IT the ability to multi-class a core component in ways that best benefit the overall infrastructure. If you need acceleration and security, just add that. If you just need security, just add that. If you need the whole package, well, this is just like [d20](#), baby, go ahead and multi-class into *everything*. It's really up to you to optimize your players (application delivery functions) in a way that makes sense for your group (organization). Much like d20, you don't have to start out multi-classed; you can add a level of "something else" later on when you decide you're ready or the group needs it. Unlike d20, some multi-classing functions can be tried out, [on an evaluation basis](#), and if it isn't a good fit for your organization you can always abandon the option and try something else.

So the next time you're evaluating your application delivery strategy – whether because you need to solve a problem and are trying to figure out where/how to add it or because you're trying to consolidate to reduce costs and improve performance – consider how "multi-classing" your load-balancer into a healing, fighting, optimized [application delivery controller](#) might achieve that goal in a cost effective, less complex solution.

The iRule Die Roller

For the truly geeky/bored, here's an [iRule](#) that implements a die roller for you – in case a game spontaneously starts in the data center and you're caught without your dice. I'm sure there's business value somewhere in knowing how to generate random values based on query parameters, but this is way more fun. And I really couldn't come up with a valid use case for business. Yet.

Usage: [http://www.example.com/roll?num=\[number of dice to roll\]&d=\[type of polyhedral to roll\]](http://www.example.com/roll?num=[number of dice to roll]&d=[type of polyhedral to roll])

Example: <http://www.nandgate.com/roll?num=3&d=6> (in case you needed to roll up stats in a hurry, standard old skool 3d6)

```
when HTTP_REQUEST {
  set uri [HTTP::uri]

  if {$uri starts_with "/roll"} {
    set v1 [URI::query [HTTP::uri] "num"]
    set v2 [URI::query [HTTP::uri] "d"]
    set total 0

    for {set i 0} { $i < $v1 } {incr i} {

      set dieroll [expr {int(rand()*$v2) + 1}]

      set result1 [expr {round($dieroll)}]
      set total [expr {$total + $result1}]
    }
    HTTP::respond 200 content "$v1 d $v2 = $total"
  }
}
```

```
}
```



- [Dear Data Center Guy](#)
- [Virtual Reality](#)
- [How to parse a URI](#)
- [It's like load balancing. On steroids.](#)
- [Picard and Dathon at El-Adrel](#)

F5 Networks, Inc. | 401 Elliot Avenue West, Seattle, WA 98119 | 888-882-4447 | f5.com

F5 Networks, Inc.
Corporate Headquarters
info@f5.com

F5 Networks
Asia-Pacific
apacinfo@f5.com

F5 Networks Ltd.
Europe/Middle-East/Africa
emeainfo@f5.com

F5 Networks
Japan K.K.
f5j-info@f5.com