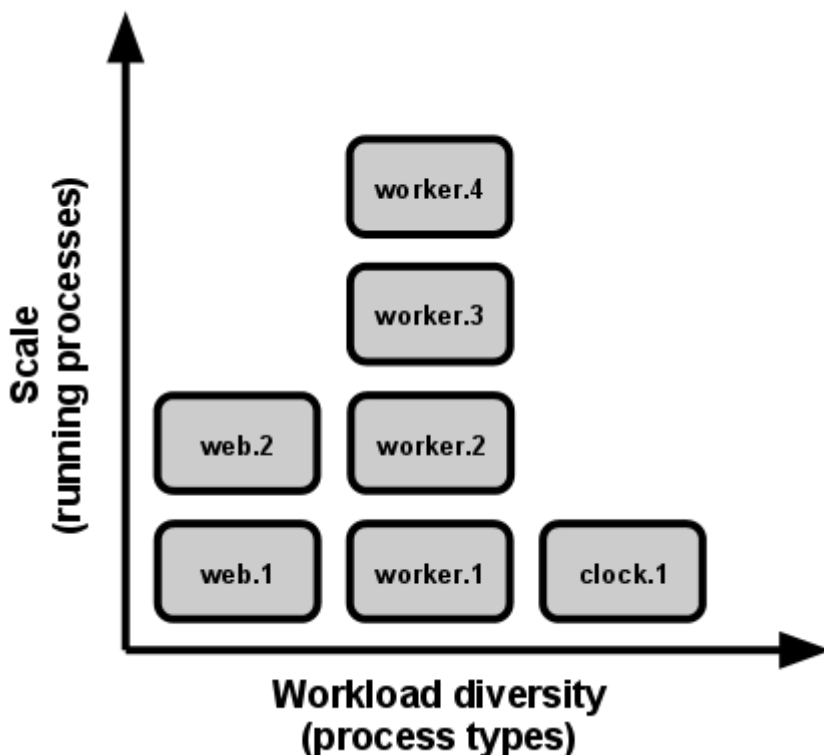


# LA APLICACIÓN DE LOS DOCE FACTORES

## VIII Concurrency

### Escalar a través del modelo de proceso

Cualquier programa de computadora, una vez ejecutado, está representado por uno o más procesos. Las aplicaciones web han tomado una variedad de formas de ejecución de procesos. Por ejemplo, los procesos PHP se ejecutan como procesos secundarios de Apache, iniciados a pedido según lo necesite el volumen de solicitud. Los procesos de Java toman el enfoque opuesto, con la JVM proporcionando un uberproceso masivo que reserva un gran bloque de recursos del sistema (CPU y memoria) al inicio, con la concurrencia administrada internamente a través de subprocesos. En ambos casos, los procesos en ejecución son mínimamente visibles para los desarrolladores de la aplicación.



En la aplicación de doce factores, los procesos son ciudadanos de primera clase. Los procesos en la aplicación de doce factores toman fuertes indicios del [modelo de proceso de Unix para ejecutar demonios de servicio](#). Con este modelo, el desarrollador puede diseñar su aplicación para manejar diversas cargas de trabajo asignando cada tipo de trabajo a un *tipo de proceso*. Por ejemplo, las solicitudes HTTP pueden ser manejadas por un proceso web, y las tareas en segundo plano de larga duración manejadas por un proceso de trabajo.

Esto no excluye procesos individuales de manejo de su propia multiplexación interna, a través de roscas en el interior del tiempo de ejecución VM, o el modelo asíncrono / evented que se encuentran en herramientas tales como [EventMachine](#), [Twisted](#), o [Node.js](#). Pero una VM individual solo puede crecer tanto (escala vertical), por lo que la aplicación también debe poder abarcar múltiples procesos que se ejecutan en múltiples máquinas físicas.

El modelo de proceso realmente brilla cuando llega el momento de escalar. La [naturaleza de compartir nada, horizontalmente divisible de los procesos de aplicaciones de doce factores](#) significa que agregar más concurrencia es una operación simple y confiable. El conjunto de tipos de procesos y el número de procesos de cada tipo se conoce como *formación de procesos*.

Los procesos de aplicaciones de doce factores [nunca deberían demonizar](#) o escribir archivos PID. En cambio, confíe en el administrador de procesos del sistema operativo (como [systemd](#) , un administrador de procesos distribuido en una plataforma en la nube o una herramienta como [Foreman](#) en desarrollo) para administrar [flujos de salida](#) , responder a procesos bloqueados y manejar reinicios y paradas iniciados por el usuario.

[ภาษาไทย \(th\)](#) | [한국어 \(ko\)](#) | [Slovensky \(sk\)](#) | [Русский \(ru\)](#) | [Français \(fr\)](#) | [日本語 \(ja\)](#) | [简体中文 \(zh\\_cn\)](#) | [Ελληνικά \(el\)](#) | [Portugués brasileiro \(pt\\_br\)](#) | [Italiano \(it\)](#) | Inglés (en) | [Turco \(tr\)](#) | [Deutsch \(de\)](#) | [Polski \(pl\)](#) | [Українська \(reino unido\)](#) | [Español \(es\)](#).

[«Anterior](#)  
[Próximo »](#)

[ภาษาไทย \(th\)](#) | [한국어 \(ko\)](#) | [Slovensky \(sk\)](#) | [Русский \(ru\)](#) | [Français \(fr\)](#) | [日本語 \(ja\)](#) | [简体中文 \(zh\\_cn\)](#) | [Ελληνικά \(el\)](#) | [Portugués brasileiro \(pt\\_br\)](#) | [Italiano \(it\)](#) | Inglés (en) | [Turco \(tr\)](#) | [Deutsch \(de\)](#) | [Polski \(pl\)](#) | [Українська \(reino unido\)](#) | [Español \(es\)](#).

Escrito por Adam Wiggins

Última actualización 2017

[Código fuente](#)

[Descargar ePub Book](#)

[Política de privacidad](#)