**MÉTODO EMERGENTE: WARP**

**NATIVE THREADS**

* Atributo de calidad: CLARIDAD
* Requerimiento: Seguir un flujo de control simple y familiar para usar llamadas simples.
* Táctica: Crear un thread nativo cada vez que se realiza una conexión, y dejar que lo maneje el SO

**ARQUITECTURA ORIENTADA A EVENTOS**

* Atributo de calidad: PERFORMANCE
* Requerimiento: Evitar la degradación de la performance.
* Táctica: Un proceso maneja múltiples conexiones y se encarga de controlar los CPU. Se invierte el control.

**UN PROCESO POR NÚCLEO**

* Atributo de calidad: PERFORMANCE
* Requerimiento: Utilizar todos los procesadores.
* Táctica: Crear varios handlers que manejen las conexiones y las llamadas a los procesadores (WORKERS) utilizando la técnica de fork (bifurcación).

**USER THREADS**

* Atributo de calidad: PERFORMANCE + CLARIDAD
* Requerimiento: Mantener el programa claro y simple dejando que las complejidades del bloquedo de entrada y salida en el trabajo multinúcleo lo maneje el compilador Glasgow de Haskel.
* Atributo de calidad: PERFORMANCE
* Requerimiento: Lograr la menor cantidad de llamadas al sistema posible.
* Táctica: Usar la llamada de Linux accept4() que es una extensión de accept y permite ahorrar dos llamadas fcntl();
* Atributo de calidad: PERFORMANCE
* Requerimiento: Minimizar el consumo de CPU
* Táctica: Crear funciones especializadas y evitar el recálculo. Uso de caché.
* Atributo de calidad: PERFORMANCE
* Requerimiento: Evitar los bloqueos innecesarios.
* Táctica: Identificar el bloqueo innecesario mediante el uso de timers para la conexión.
* Atributo de calidad: PERFORMANCE
* Requerimiento: La estructura String es una lista de caracteres Unicode y a nivel de performance de servidores es muy lenta y Unicode es muy complejo desde que el protocolo HTTO est{a basado en flujos de bytes.
* Táctica: Utilizar otra estructura (ByteString) que es un array de bytes con metadata.
* Atributo de calidad: SEGURIDAD
* Requerimiento: Evitar los ataques Slowloris (Muchas conexiones desde un mismo hardware con bajos niveles de transferencias).
* Táctica: Utilizar los timers de conexión para saber si en cada conexión se está recibiendo o no más datos. Si no se recibe más datos la conexión se corta.
* Atributo de calidad: PERFORMANCE
* Requerimiento: Cuando existe gran nivel de concurrencia Warp da buenos resultados, pero cuando la concurrencia es 1, la respuesta es muy lenta.
* Táctica: Cuando es posible enviar encabezado y cuerpo en un único paquete TPC.