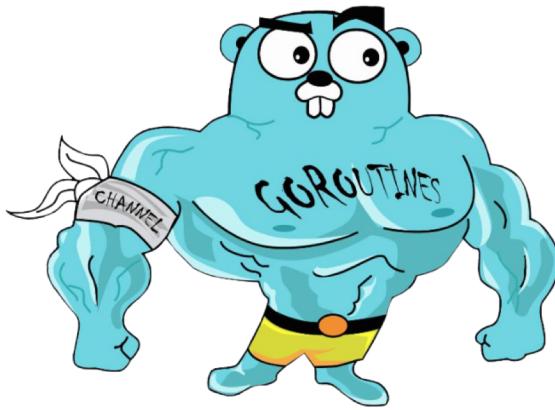


<WE/BUILD>



# GOROUTINE UNDER THE HOOD

Bien Vo - @GolangVietNam





# AGENDA

<WE/BUILD>



## GOROUTINE



## GO SCHEDULER



## SUMMARY





GOROUTINE

<WE/BUILD>

# GOROUTINE





# GOROUTINE

<WE/BUILD>

“A GOROUTINE is a lightweight thread managed by the Go runtime.”

- GoTour -

“A GOROUTINE is a function or method which executes independently and simultaneously in connection with any other GOROUTINE present in your program.”

- Geeksforgeeks -

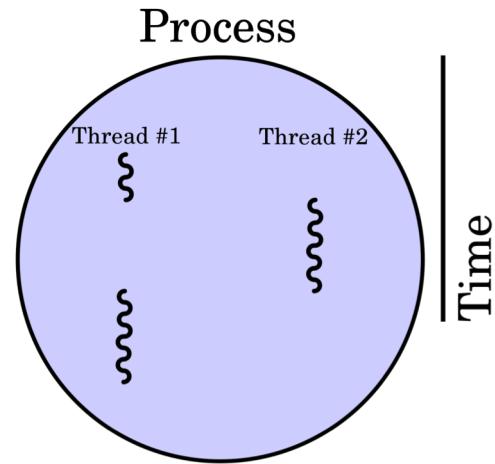




# GOROUTINE

<WE/BUILD>

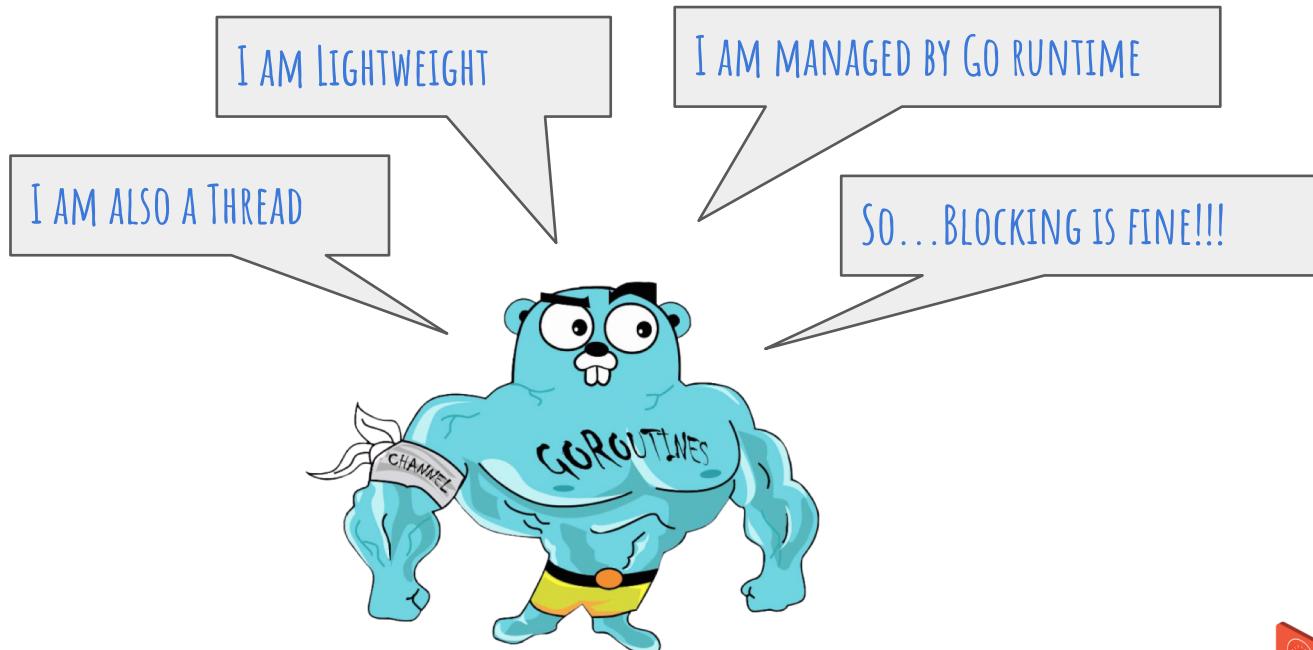
A thread is just a sequence of instructions that can be executed independently by a processor





# GOROUTINE

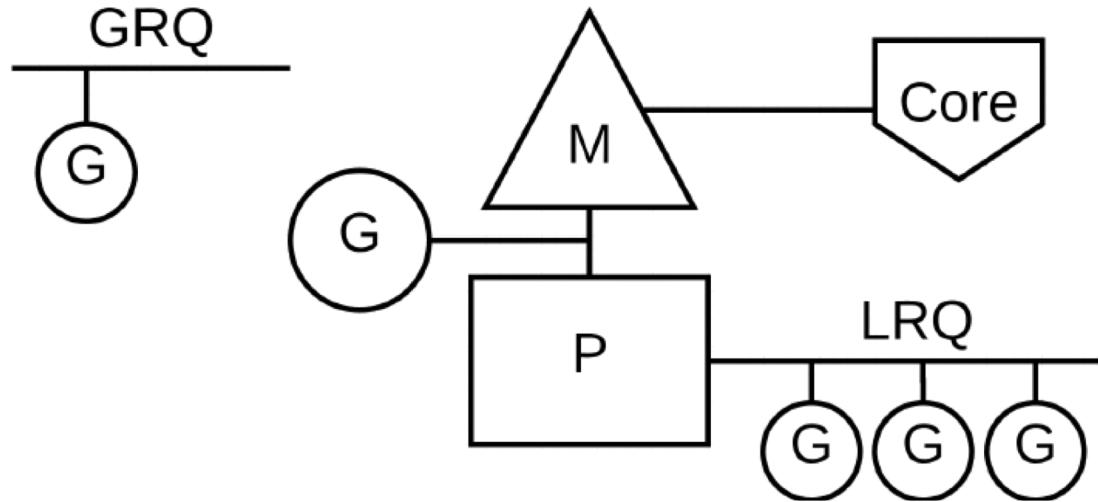
<WE/BUILD>





# GOROUTINE

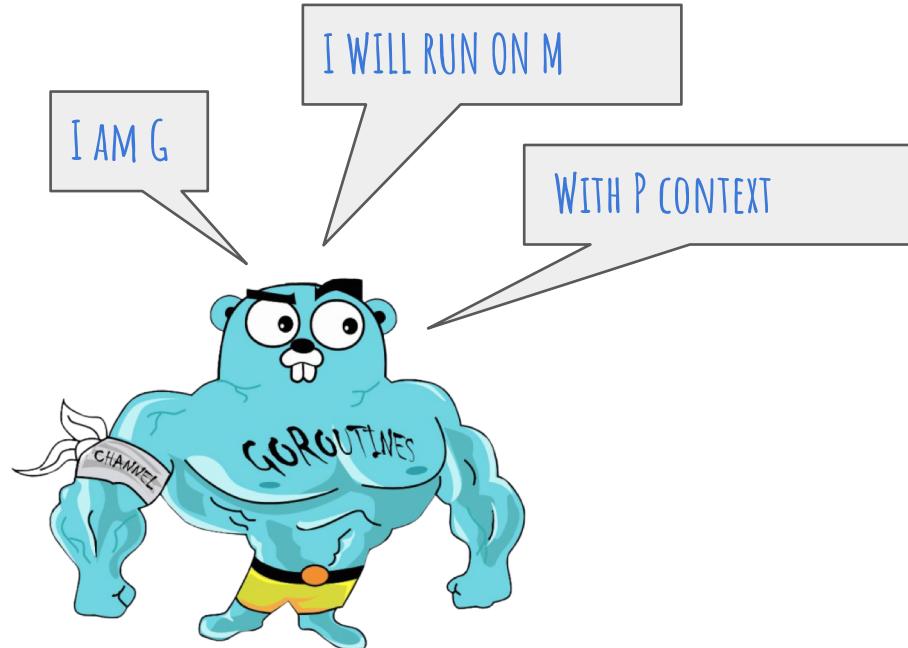
<WE/BUILD>





# GOROUTINE

<WE/BUILD>





GOROUTINE

<WE/BUILD>

# GO SCHEDULER





# GOROUTINE SCHEDULER

<WE/BUILD>

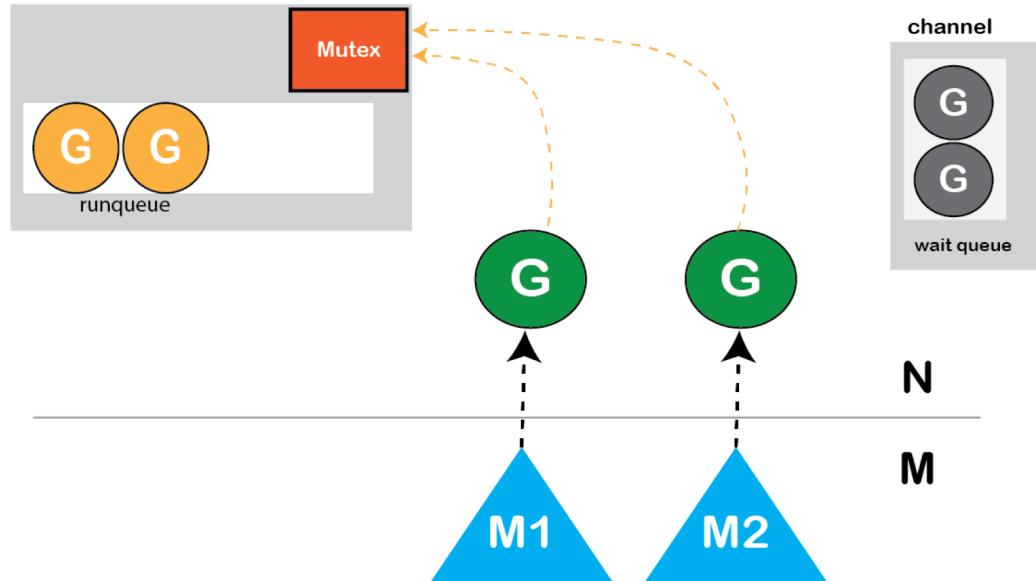
PREEMPTIVE OR COOPERATIVE ??????





# GOROUTINE SCHEDULER - BLOCKING BY CHANNEL

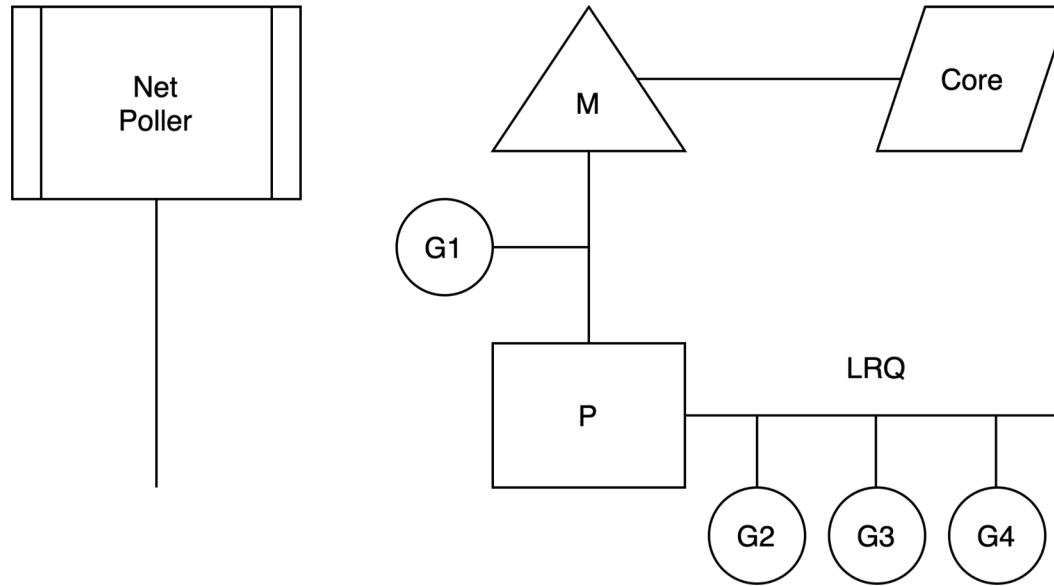
<WE/BUILD>





# GOROUTINE SCHEDULER - ASYNCHRONOUS SYSTEM CALLS

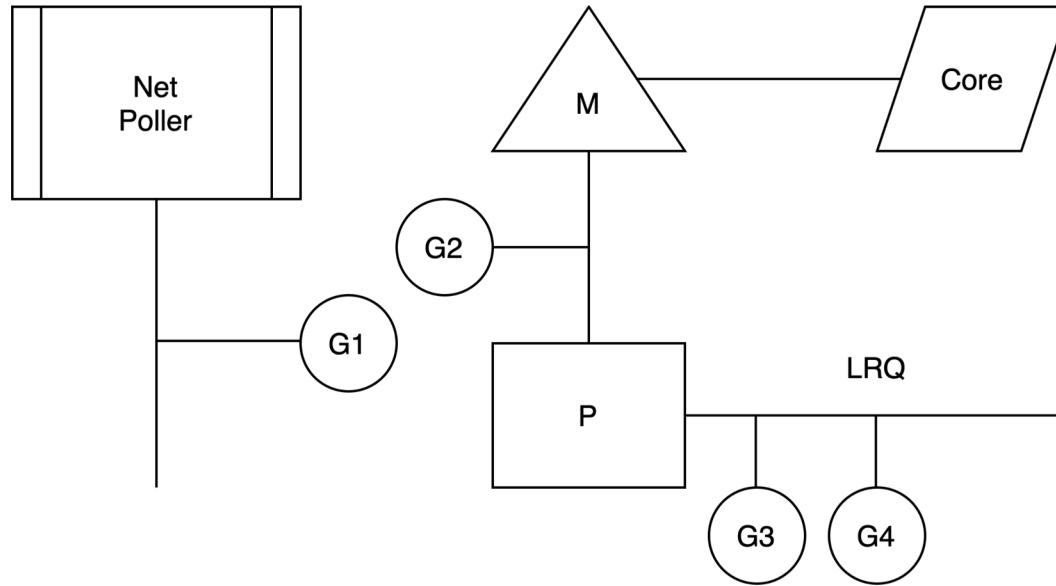
<WE/BUILD>





# GOROUTINE SCHEDULER - ASYNCHRONOUS SYSTEM CALLS

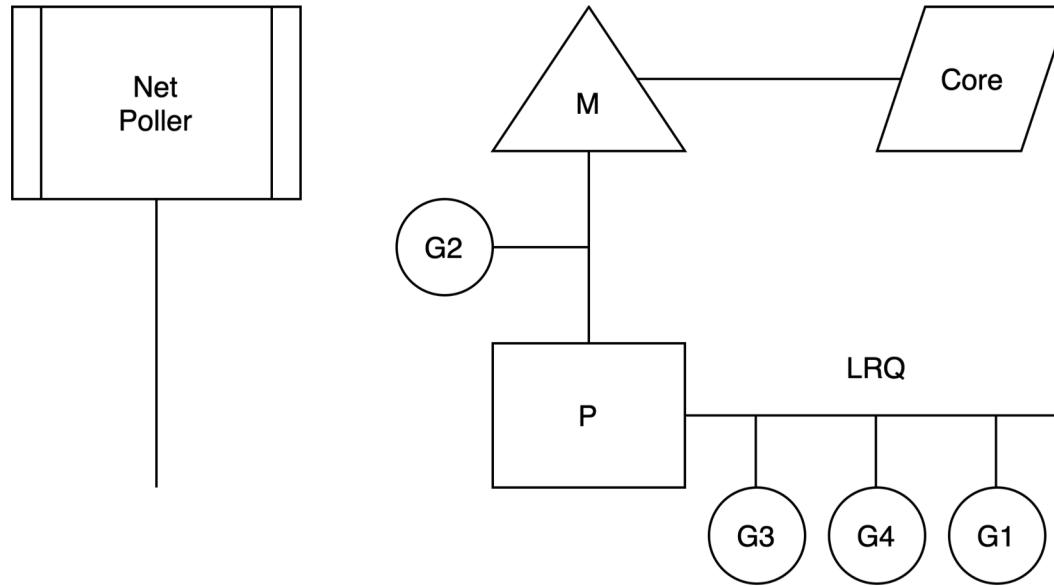
<WE/BUILD>





# GOROUTINE SCHEDULER - ASYNCHRONOUS SYSTEM CALLS

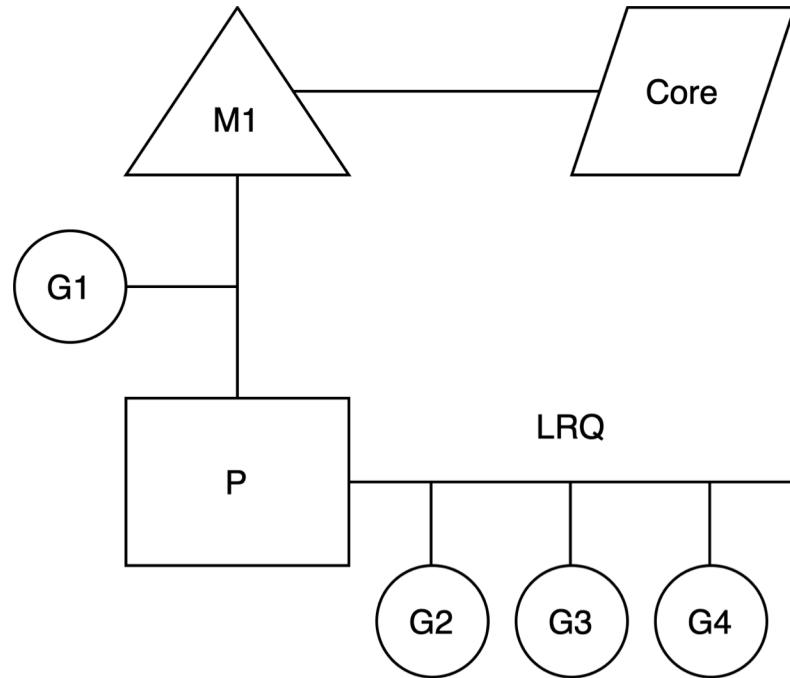
<WE/BUILD>





# GOROUTINE SCHEDULER - SYNCHRONOUS SYSTEM CALLS

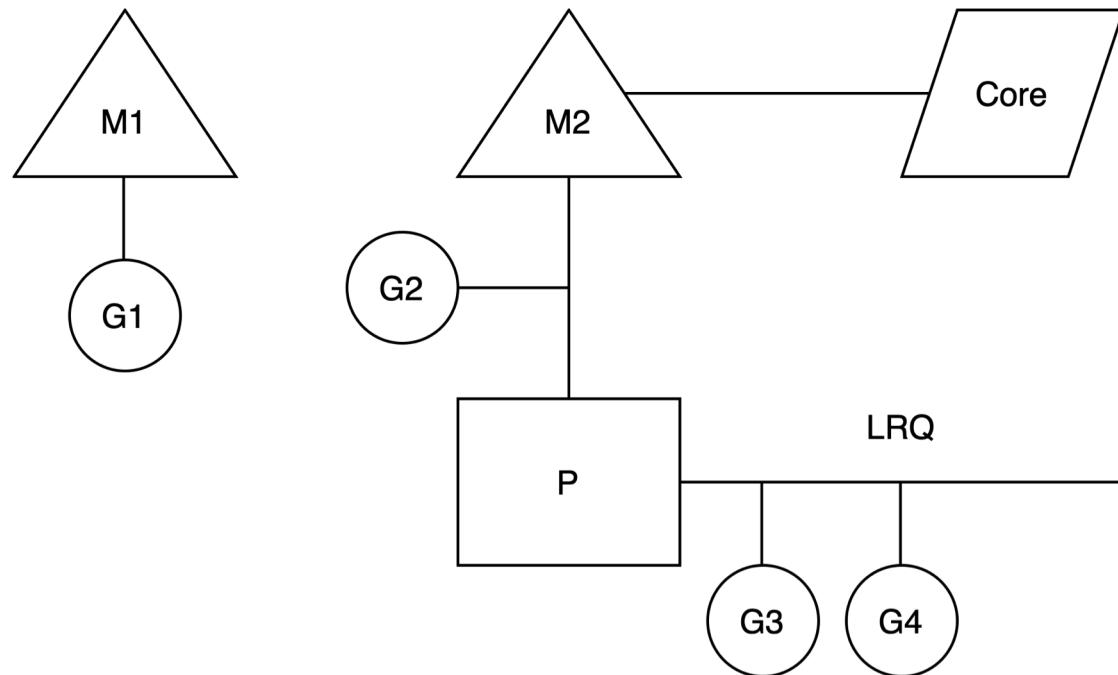
<WE/BUILD>





# GOROUTINE SCHEDULER - SYNCHRONOUS SYSTEM CALLS

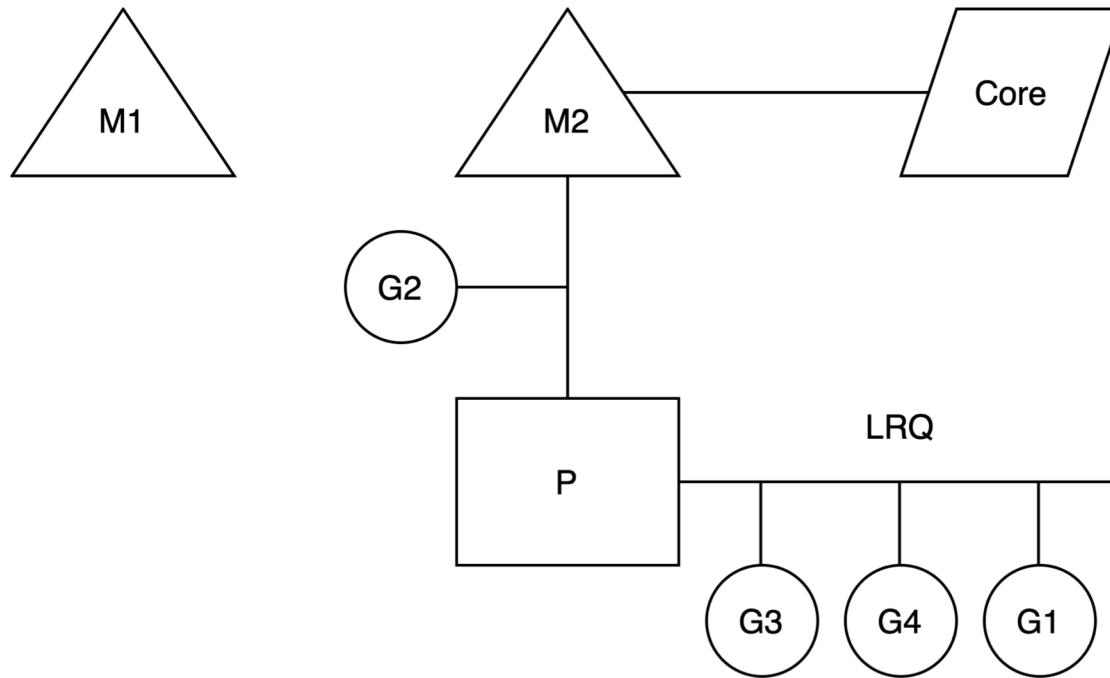
<WE/BUILD>





# GOROUTINE SCHEDULER - SYNCHRONOUS SYSTEM CALLS

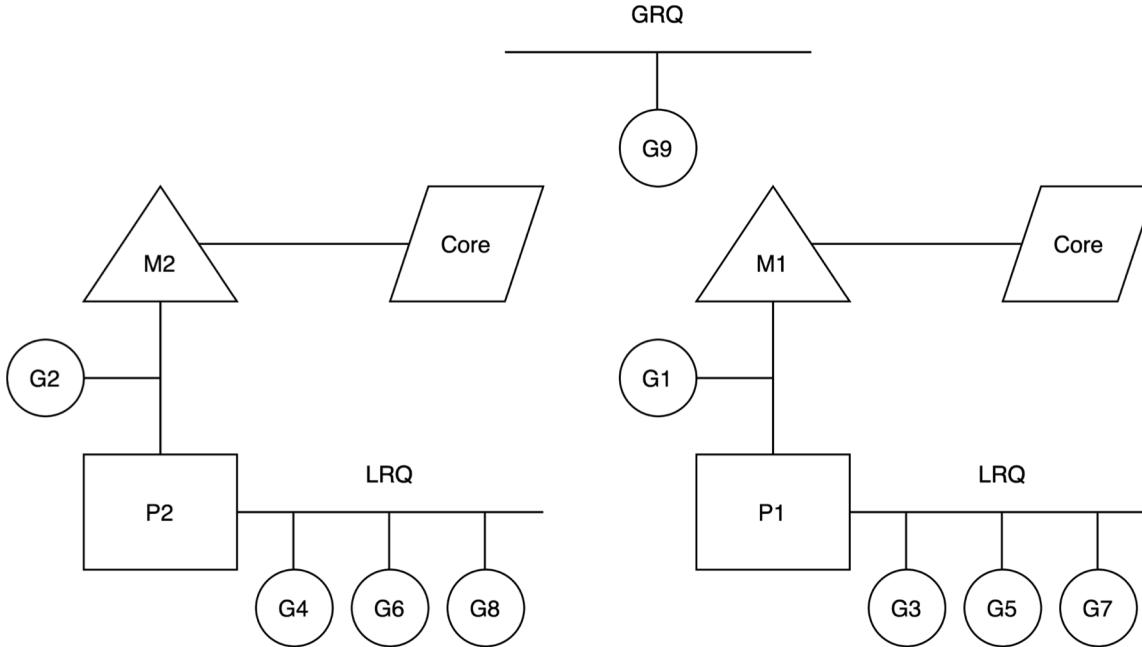
<WE/BUILD>





# GOROUTINE SCHEDULER - WORK STEALING

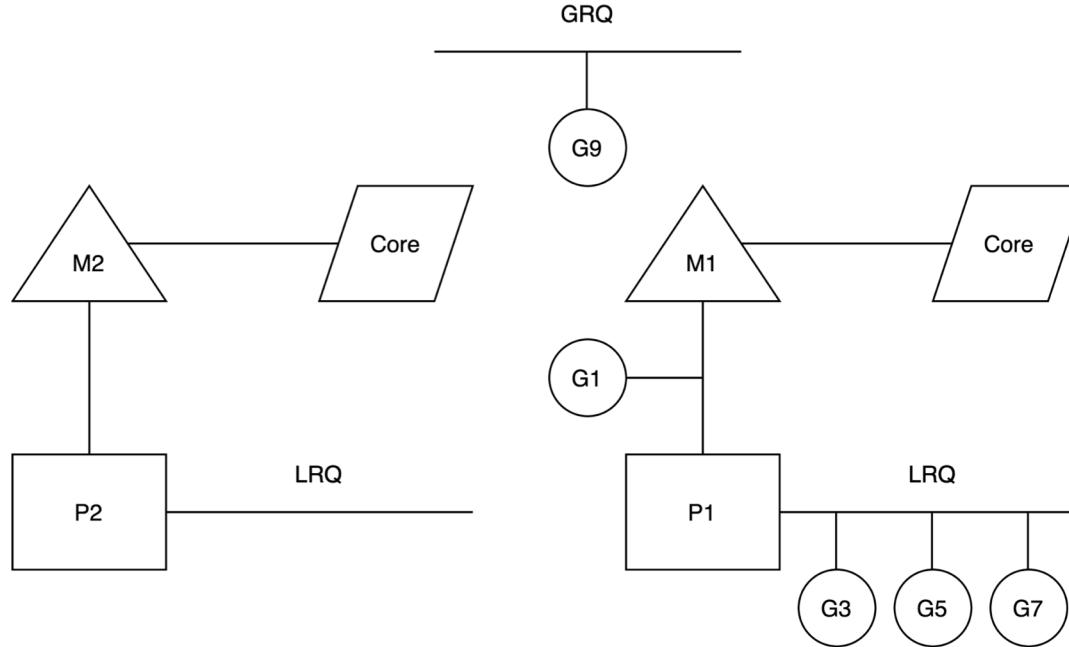
<WE/BUILD>





# GOROUTINE SCHEDULER - WORK STEALING

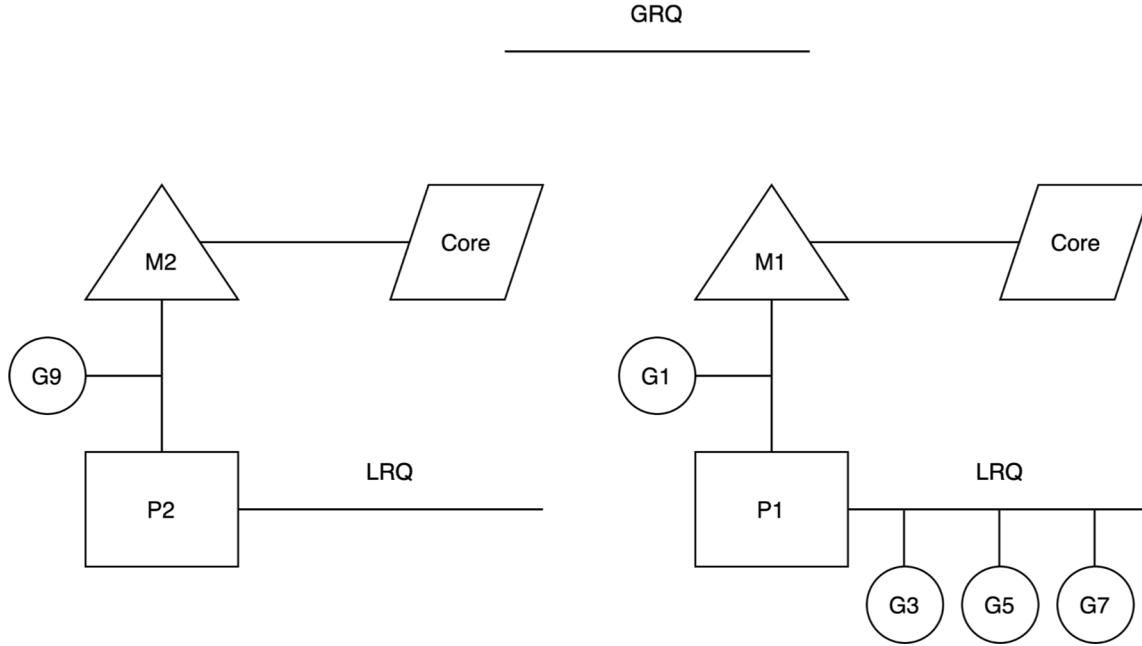
<WE/BUILD>





# GOROUTINE SCHEDULER - WORK STEALING

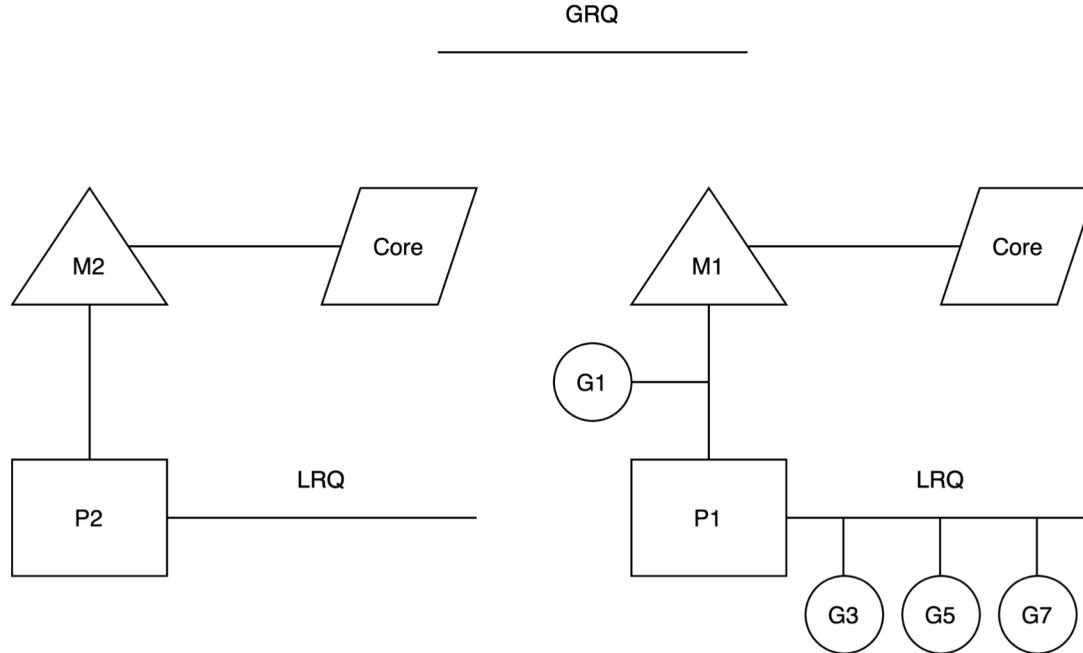
<WE/BUILD>





# GOROUTINE SCHEDULER - WORK STEALING

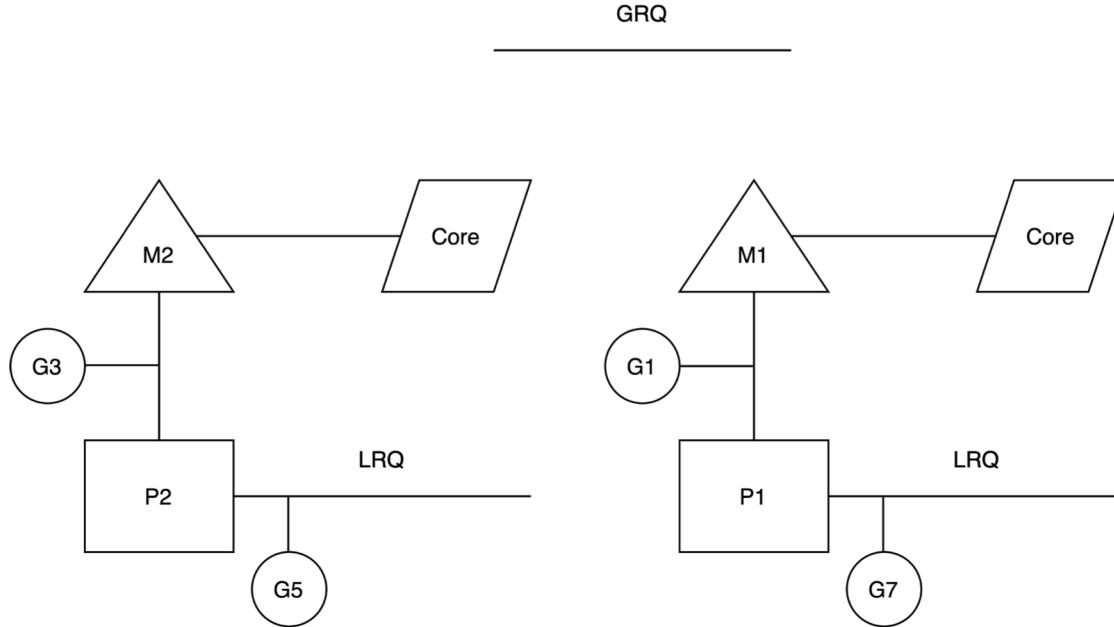
<WE/BUILD>





# GOROUTINE SCHEDULER - WORK STEALING

<WE/BUILD>





## SUMMARY

<WE/BUILD>

A **GOROUTINE** is a lightweight thread managed by the Go runtime.

Instead of OS scheduler, Go Scheduler is responsible for **GOROUTINE** management.

Network poller helps Go runtime handle Goroutine with Asynchronous System Calls

Golang takes advantage of resource by using Work stealing mechanism





Q&A

<WE/BUILD>





<WE/BUILD>

# THANK YOU!

