

Documentation Ipanema

Baptiste Pires

April 8, 2024

1 Introduction

2 Structures

Dans cette section nous allons voir les différentes structures utilisées dans Ipanema et leurs utilités

2.1 struct sched_ipanema_entity

Nous allons commencer par la structure qui représente une entité d’ordonnancement. La structure est déclarée de cette façon :

```
1 struct sched_ipanema_entity {
2     union {
3         struct rb_node node_runqueue;
4         struct list_head node_list;
5     };
6
7     /* used for load balancing in policies */
8     struct list_head ipa_tasks;
9
10    int just_yielded;
11    int noproempt;
12
13    enum ipanema_state state;
14    struct ipanema_rq *rq;
15
16    /* Policy-specific metadata */
17    void *policy_metadata;
18
19    struct ipanema_policy *policy;
20 };
```

Les deux premiers champs, `struct rb_node node_runqueue` et `struct list_head node_list` sont utilisés pour maintenir une tâche dans une runqueue. Dû à l’union, on peut être dans une seule runqueue à un moment donné (soit dans une runqueue qui se sert d’un arbre rouge-noir comme CFS, ou une runqueue qui utilise une liste comme un algorithme FIFO par exemple).

Le champ `struct list_head ipa_tasks` est utilisé par les différentes politiques d’ordonnancement pendant l’équilibrage de charge. Par exemple, si on migre des tâches d’un cœur C_1 à un cœur C_2 , on a besoin de retirer les tâches de la runqueue de C_1 en ayant le verrou sur celle-ci. Quand on retire les tâches on peut les ajouter à la liste `ipa_tasks`. Une fois qu’on a retiré toutes les tâches qu’on souhaite migrer, on prend le verrou sur la runqueue de C_2 et on peut itérer sur la liste que l’on vient de construire.

`int just_yield` permet de savoir si on vient d’appeler `sched_yield()` ou non.

`enum ipanema_state state` représente l’état d’une entité Ipanema. Les états possibles sont les suivants :

```

1 enum ipanema_state {
2     IPANEMA_NOT_QUEUED, /* La tache n'est pas dans une runqueue */
3     IPANEMA_MIGRATING, /* La tache est en train de migrer d'un coeur a un autre */
4     IPANEMA_RUNNING, /* La tache est dans une runqueue et s'execute */
5     IPANEMA_READY, /* La tache est dans une runqueue mais ne s'execute pas */
6     IPANEMA_BLOCKED, /* La tache est bloquee (I/O, lock, etc...) */
7     IPANEMA_TERMINATED, /* La tache meurt (?) */
8     /*
9      * A special state that is equivalent to IPANEMA_READY, but makes it
10     * possible to figure out that the state change came from tick().
11     */
12     IPANEMA_READY_TICK
13 };

```

`struct ipanema_rq *rq` c'est un pointeur vers la `struct ipanema_rq` sur laquelle la tâche s'exécute (ou va s'exécuter).