

## Aufgabe 8 OCaml (Threads): P2P-Server

[23 Punkte]

In dieser Aufgabe möchten wir ein Peer-to-Peer-Netzwerk simulieren. Jeder Nutzer kann für einen Schlüssel Daten bereitstellen. Ein Broker verwaltet dabei wer für welchen Schlüssel Daten anbietet und leitet Anfragen weiter. Wir definieren einen Typ

```
type ('a,'b) t = Publish of 'a * 'b channel
              | Request of 'a * 'b event option channel
```

wobei **Publish** aus dem Schlüssel und einem Channel besteht auf dem die Daten fortlaufend geschickt werden, und **Request** aus dem Schlüssel und einem Channel besteht auf dem ein Event zum Empfangen der Daten zurückgeschickt wird, falls jemand dazu Daten veröffentlicht hat.

Implementieren Sie die folgenden Funktionen:

1. [11 Punkte] `val broker : unit -> ('a,'b) t channel` wobei `broker ()` einen neuen Broker startet, der Anfragen für Schlüssel vom Typ `'a` mit Daten vom Typ `'b` verwaltet.
2. [7 Punkte] `val publish : ('a,'b) t channel -> 'a -> 'b -> unit` wobei `publish b k v` einen Thread startet, der die Daten `v` dauerhaft bereitstellt und dies beim Broker `b` unter dem Schlüssel `k` veröffentlicht.
3. [5 Punkte] `val request : ('a,'b) t channel -> 'a -> 'b option` wobei `request b k` bei einem Broker `b` den Schlüssel `k` anfragt und das Ergebnis empfängt (sofern vorhanden).

*Hinweis:* Sie dürfen für Ihre Lösung annehmen, dass jeder Schlüssel nur einmal veröffentlicht wird.