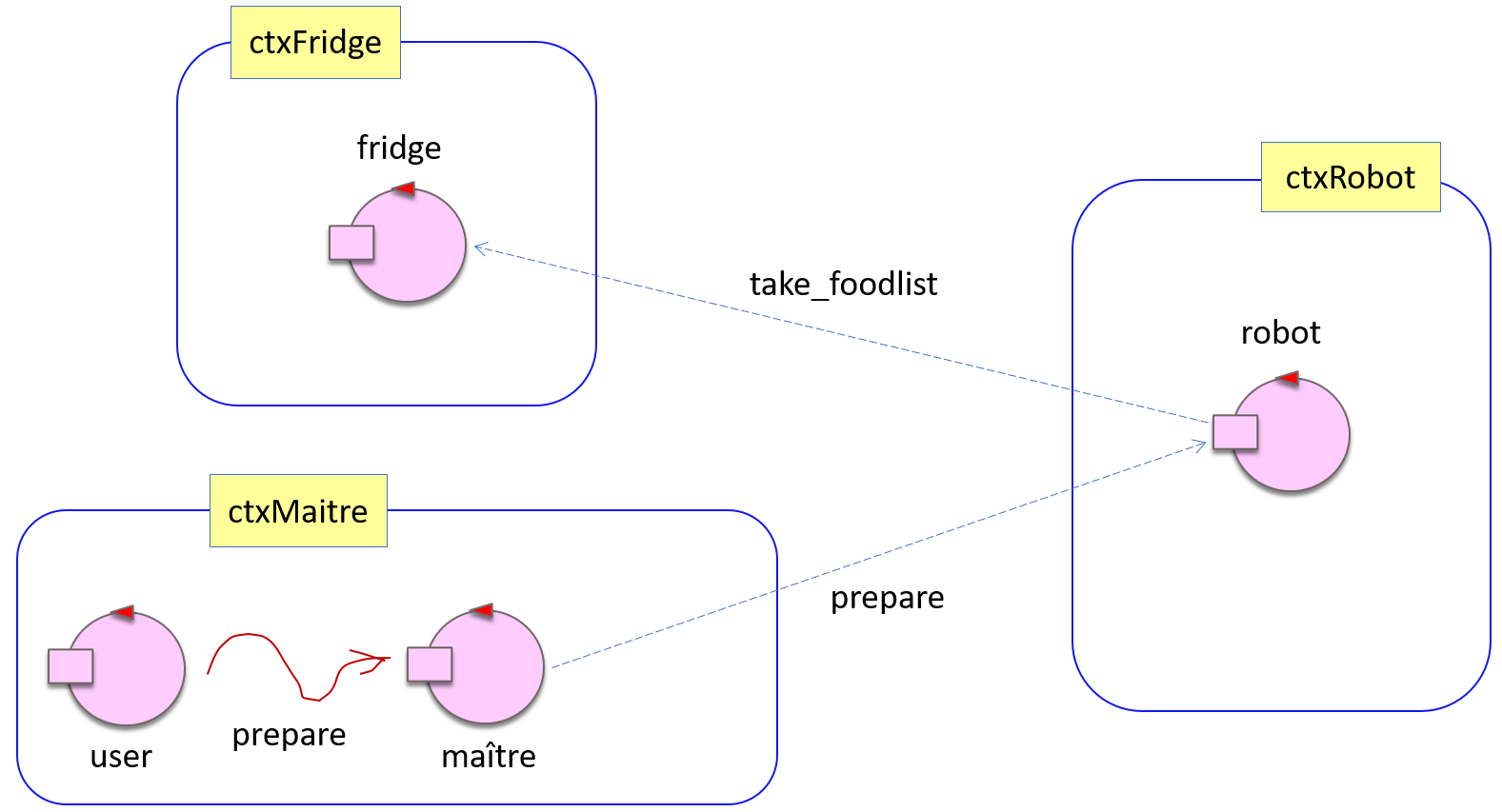
# Sprint 3 – Prepare

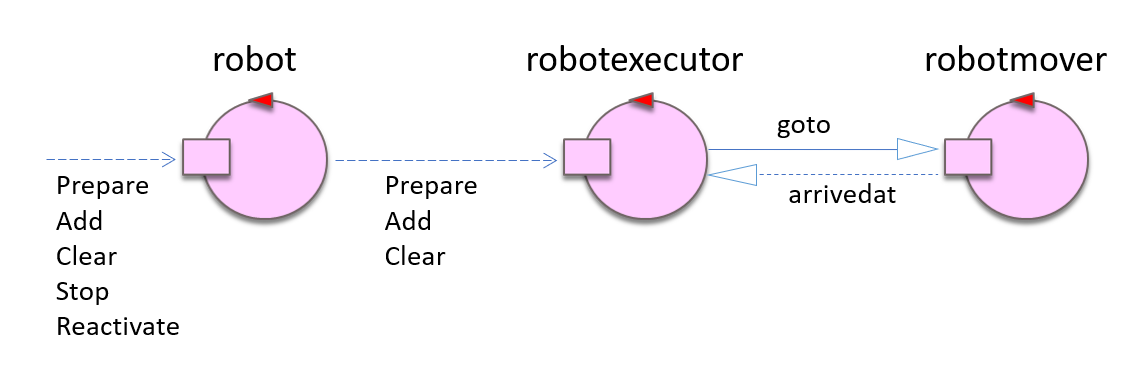
La Prepare è il task che permette di ordinare al robot di preparare la stanza per il buffet, ovvero prendere le stoviglie dalla dispensa e il cibo dal frigo e posizionarli sul tavolo (la lista di stoviglie e cibo è predeterminata e si considera precaricata sul robot nel file preparerequirements.pl).



L’interazione tra le entità del sistema è rappresentata in figura:



Perché il robot sia in grado di svolgere autonomamente il task, senza perdere la capacità di ricevere altri comandi dall’esterno, si rende necessario strutturarlo secondo un’architettura a layer, formata da un attore di alto livello che riceve i comandi dal maître e li inoltra a componenti più interne: introduciamo quindi il robotexecutor, che gestisce in autonomia le diverse fasi di un task, e il robotmover, che si occupa del movimento. Quest’ultima, in questa fase, è ancora un’entità logica, che nasconde il funzionamento reale del movimento del robot fisico o virtuale.

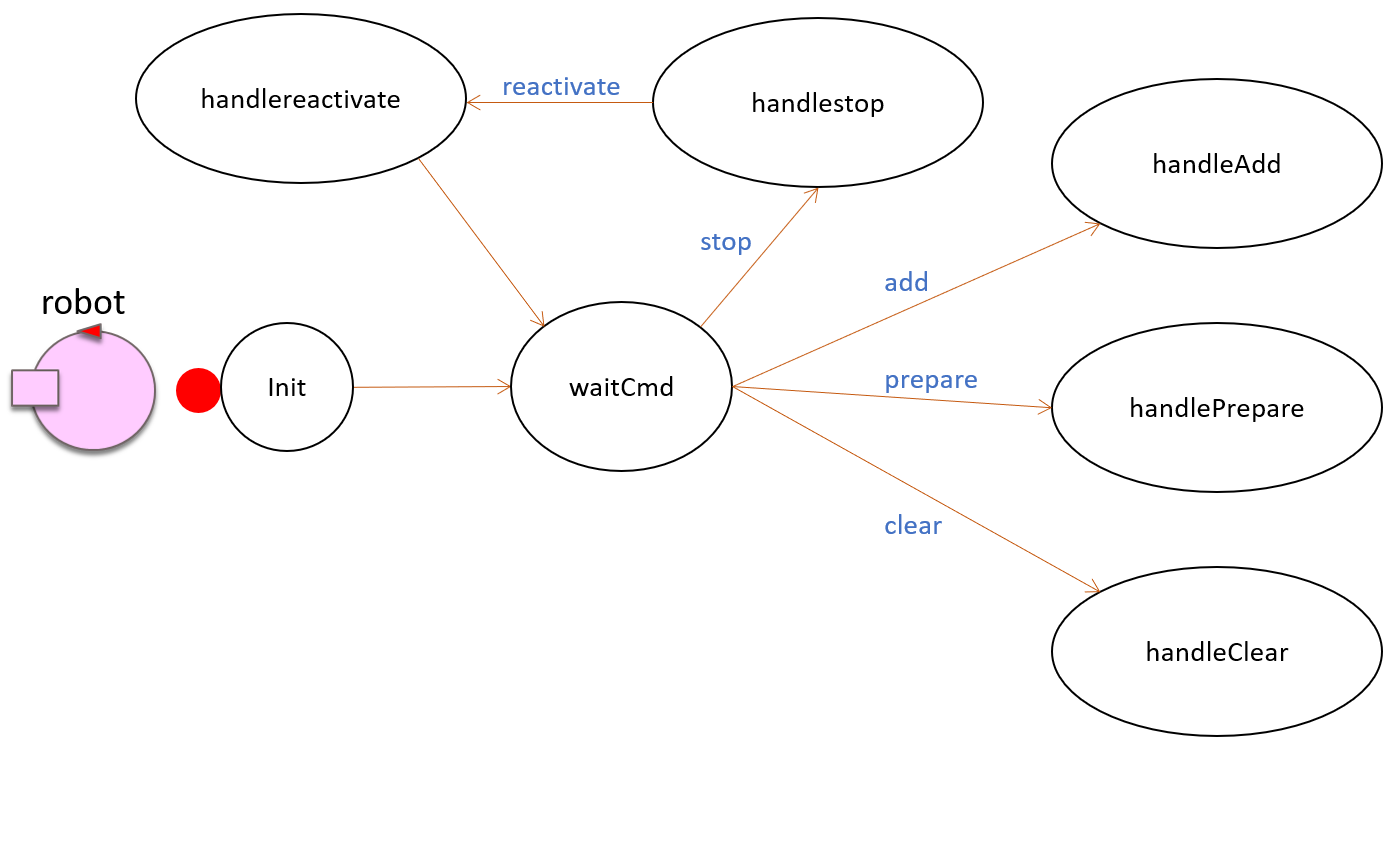


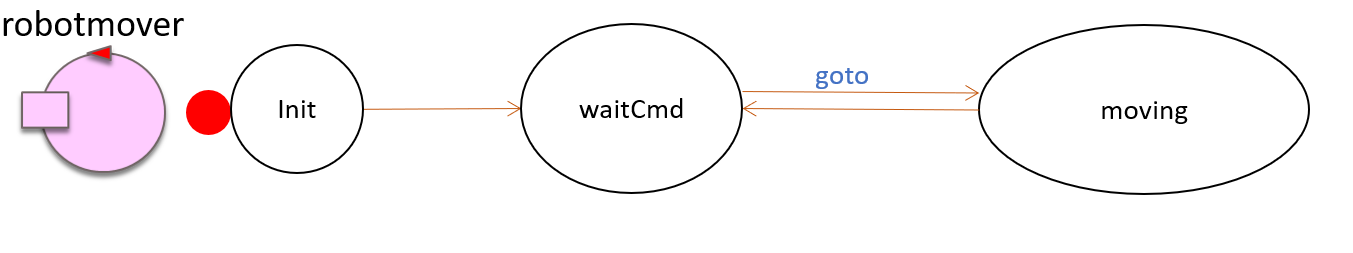
L’interazione maître-robot e robot-robotexecutor è realizzata con messaggi di tipo Dispatch, mentre l’interazione robotexecutor-robotmover è realizzata con messaggi di tipo Request/Reply con il seguente id:

* goto(Destination): che permette di comunicare la destinazione desiderata
* arrivedat(Destination): con cui il robotmover comunica l’avvenuto spostamento

L’utilizzo di messaggi Request/Reply consente al robotmover di essere un componente indipendente e disaccoppiato dalla logica specifica del butler, e quindi anche riutilizzabile.

Di seguito il dettaglio del comportamento dei tre attori, esprimibile come Finite State Machine:





Per il robotexecutor il dettaglio riguarda solamente l’esecuzione del task prepare, oggetto dello sprint:

