This work presents a parallel numerical strategy to transport Lagrangian particles in a fluid using a dynamic load balance strategy. Both fluid and particle solvers are parallel, with two levels of parallelism. The first level is based on a substructuring technique and uses message passing interface (MPI) as the communication library; the second level consists of OpenMP pragmas for loop parallelisation at the node level. When dealing with transient flows, there exist two main alternatives to address the coupling of these solvers. On the one hand, a single-code approach consists in solving the particle equations once the fluid solution has been obtained at the end of a time step, using the same instance of the same code. On the other hand, a multi-code approach enables one to overlap the transport of the particles with the next time-step solution of the fluid equations, and thus obtain asynchronism. In this case, different codes or two instances of the same code can be used. Both approaches will be presented. In addition, a dynamic load balancing library is used on the top of OpenMP pragmas in order to continuously exploit all the resources available at the node level, thus increasing the load balance and the efficiency of the parallelisation and uses the MPI.