The basic application is using Linux system as a software router, where Linux kernel serves as forwarding plane, while BIRD serves as a control plane, communicating with other routers in the network, discovering the network topology and computing the routing table. BIRD consist of **bird** daemon and **birdc** interactive CLI client used for supervision. BIRD supports for multiple protocol instances and multiple routing tables. Unlike FRR, where each protocol is a separate process, in BIRD all protocols are handled by one process [10]. There are few basic conceptual objects in BIRD, including:

- Routes
  Routes consist of destination network prefix and an associated interface. Routes generated by protocols and stored in routing tables.

- Protocols
  Protocols objects represent instances of routing protocols (e.g., OSPF, BGP, RIP)or other route sources (static, kernel). Protocols generate routes, propagate them to routing table and also receive routes form it.

- Routing tables
  Routing tables are data tables where routes are accumulated. Protocols export network routes to the routing table and also can import routes from the routing table to protocols. This way information about routes is exchanged among different routing protocols. BIRD routing tables are not automatically synchronized with kernel forwarding table. **Kernel** protocols is in charge to connect BIRD routing table to kernel forwarding table. There is also a **pipe** protocol to exchange routes between two routing tables. Figure 2.4 illustrate the relation of BIRD objects mentioned above.

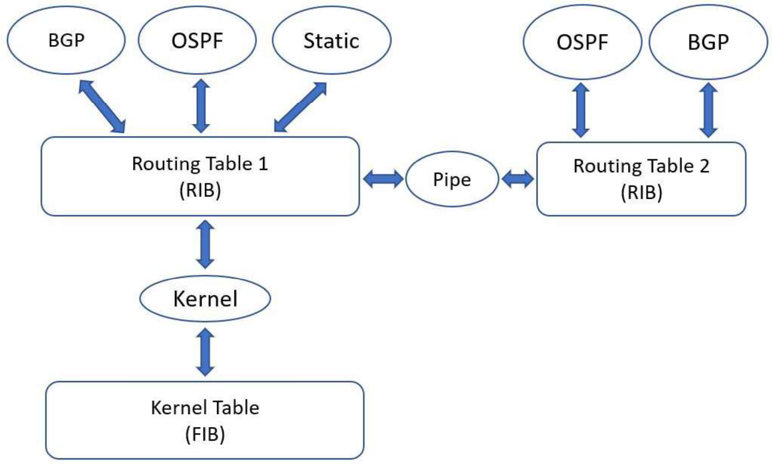

Figure 2.4 Illustration of relation of BIRD objects.

BIRD supports following features:

- IPv4 and IPv6
- Multiple routing tables
- BGP
- OSPF
- BFD
- Babel
- Route filtering

2.4 Convergence Time

The routing layer determines which path is the most efficient through network with routing algorithm. If a link in the network down the routers have to re-route to another path. Convergence time is the time that is required for the routers in a network to learn about routers in the given network. This time is important because it helps network administrator to determine the amount of time will take for a network to recover in the event of link failure between routers. An OSPF convergence process consist of following major operation [11] :

- Failure detection. Failures in a network can be detected with the help of hello protocol. The Hello Protocol is responsible for initiating and sustaining neighbor relationships. It also ensure that communication between neighbor is bidirectional. Hello message are sent periodically out all interface over broadcast network.

- Event propagation. In OSPF topology changes (event) are advertised by means of LSA flooding. In general LSA propagation time is determined by the following factors :

  1. LSA generation delay, is the time it takes to generate LSA after a link failure detected.
  2. LSA reception delay, is a sum of the ingress queueing delay and LSA arrival delay
  3. Packet propagation, is the time to propagate LSA packet to the router in the network.

- Shortest Path First (SPF) calculation. The process to execute Djikstra algorithm finding the shortest path to destination. The computational complexity of SPF algorithm can be bounded as *O(n log(n))* where *n* is the number of nodes [11].

- RIB/FIB update. The time to needed to update routing table and forwarding table.

Based on the operation mentioned above the convergence time of OSPF can be expressed as :

$$ConvergenceTime = FailureDetection + EventPropagation + SPFcalculation + RIB/FIBupdate \tag{2.1}$$

2.5 Quality of Service

According to ETSI [12] there are several benchmark to measure the performance of NFVI related to communication aspect. The benchmark includes:

- Throughput
  Througput is a key benchmark for the NFVI. This information will let us know the maximum capacity of a system under test (SUT). It can assist capacity planning and network engineering of the production of the interested