Sharif University of Technology

Computer Engineering Department

Software-Defined Networking

Ali Movaghar

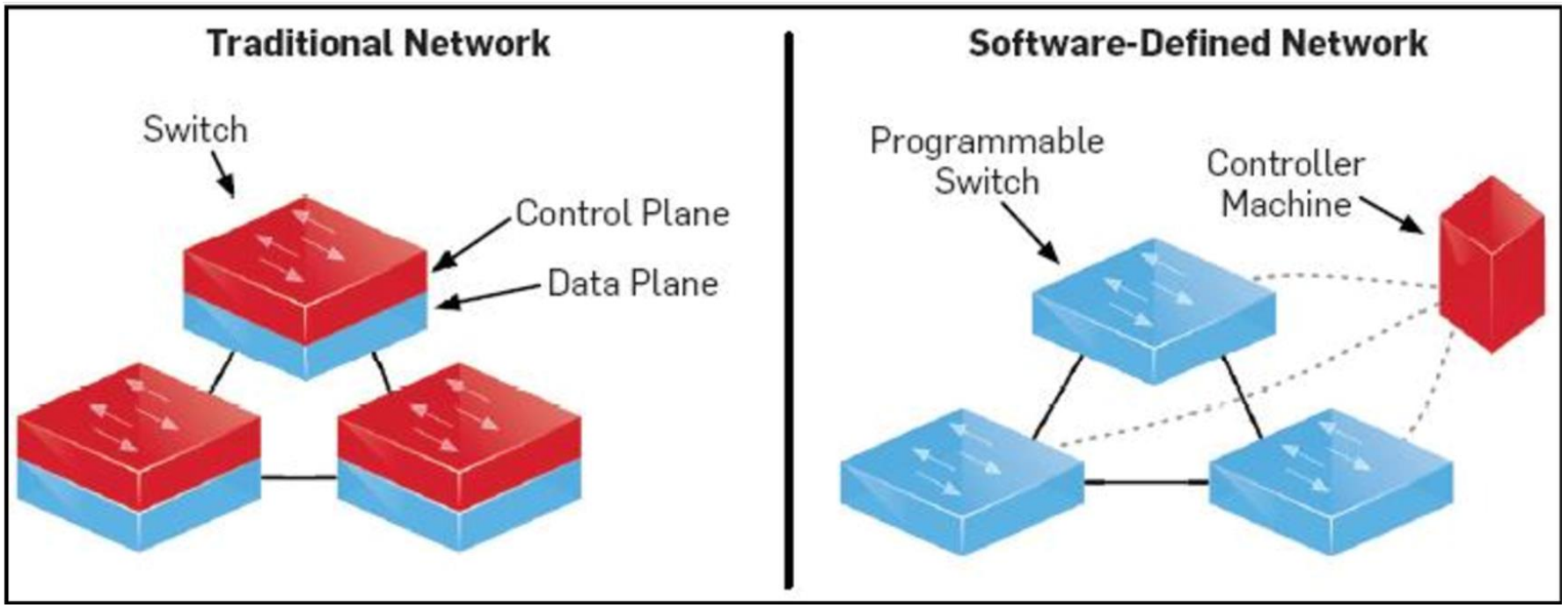
Mohammad Hosseini

TA: Iman Rahmati & Farbod Shahinfar

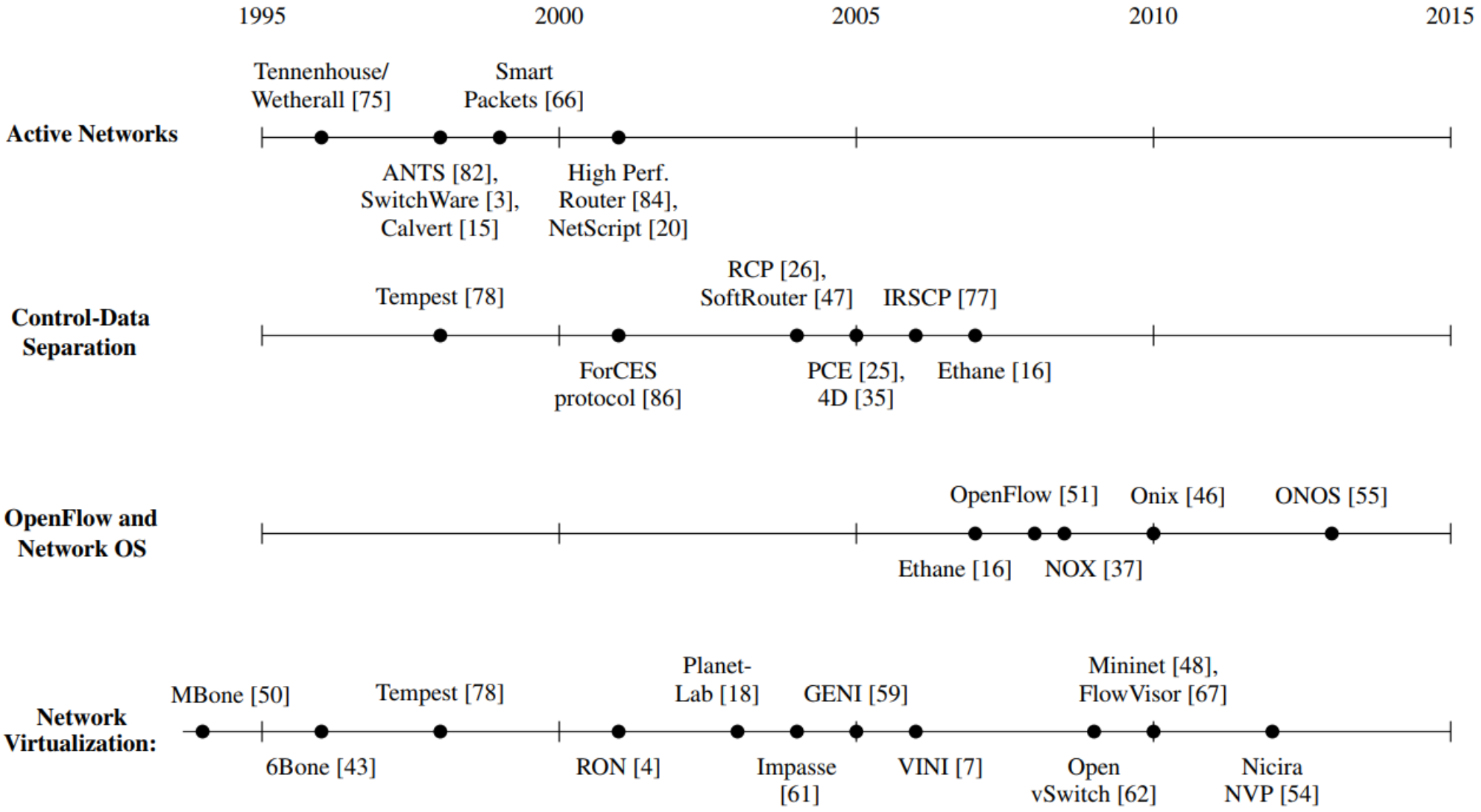
Historical Research Background

Includes slides from courses taught by Mohammad Alizadeh (MIT), Jennifer Rexford (Princeton), and Nick McKeown (Stanford).

Historical Research Background of Software-Defined Networking

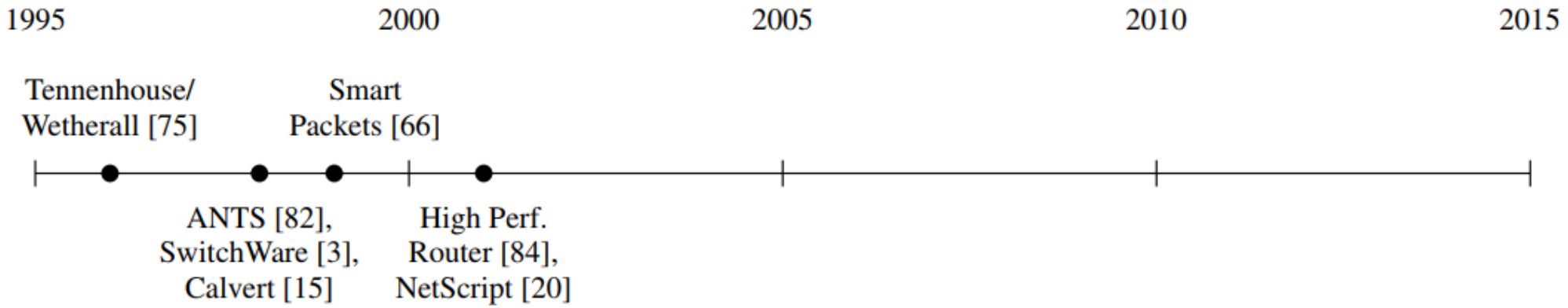


The Road to SDN



The road to SDN: an intellectual history of programmable networks,
ACM SIGCOMM Computer Communication Review 44.2 (2014)

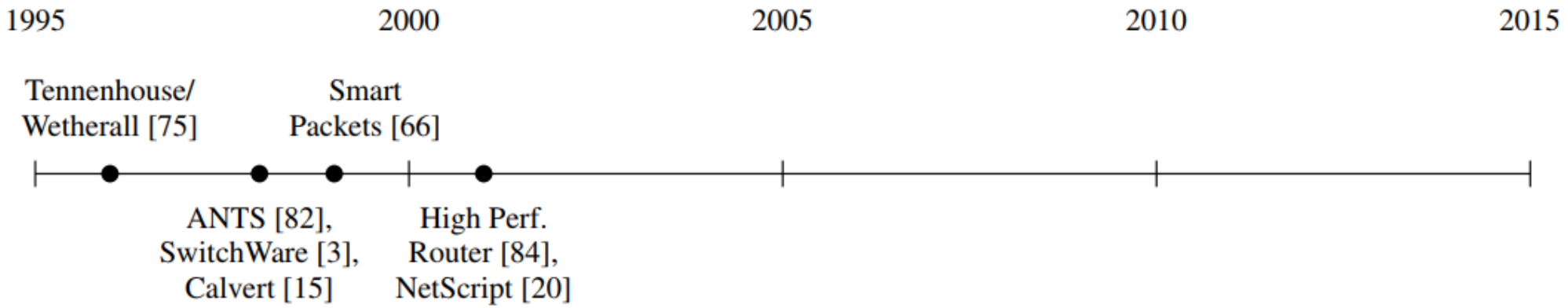
Active Networks (1)



- The need to open up network control
- **Active Networking**: A programming interface supporting the construction of custom functionality to apply to a subset of packets passing through the node

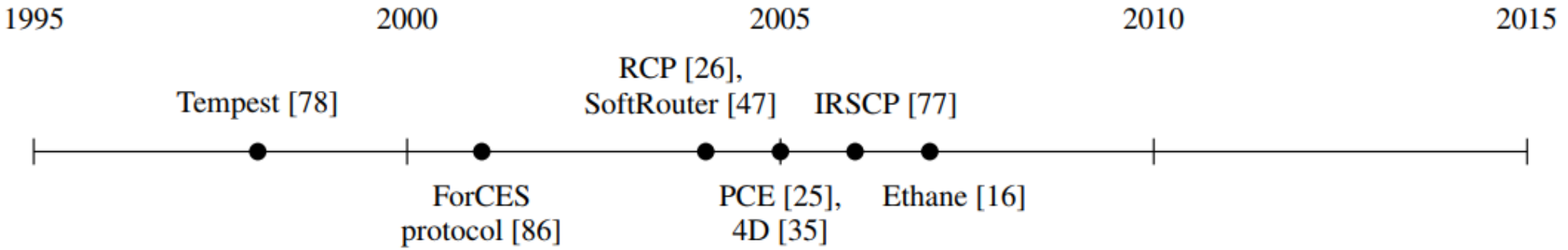
- [3] The SwitchWare active network architecture
- [82] A toolkit for building and dynamically deploying network protocols
- [66] Smart packets for active networks
- [84] Design issues for high performance active routers
- [20] The NetScript active network system

Active Networks (2)



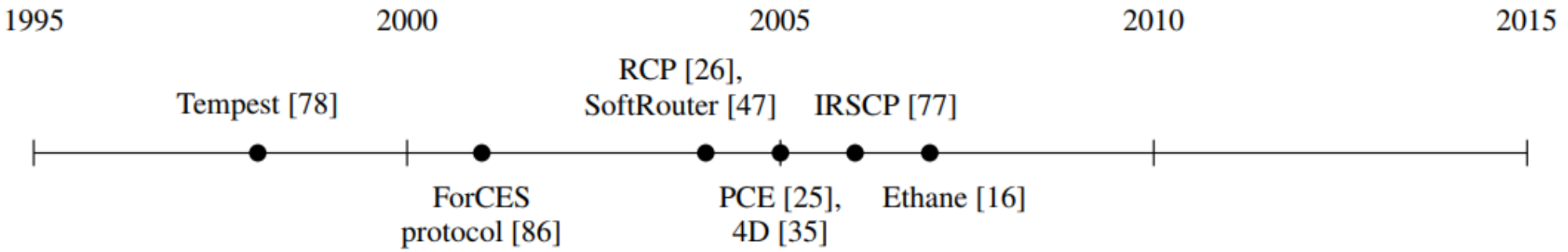
- Two programming models:
  - 1- **The capsule model**: the code to execute at the nodes was carried in-band in data packets [82]
  - 2- **The programmable router/switch model**: the code to execute at the nodes was established by out-of-band mechanism
- Performance and security issues
- Lack of an immediately compelling problem

Separating Control and Data Planes (1)



- Network operators sought better approaches to network management and control, especially the control over the paths used to deliver traffic (traffic engineering).
- The means for performing traffic engineering using conventional routing protocols were primitive.
- The tight integration between the control and data planes made network management and control tasks such as controlling routing behavior, exceedingly challenging.

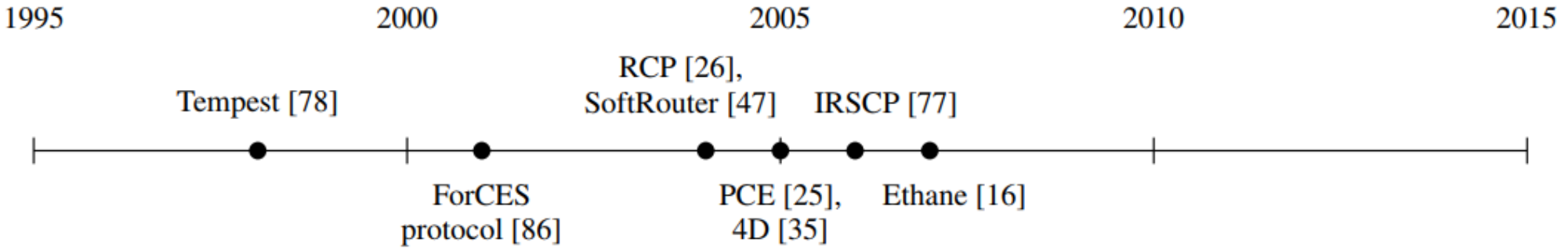
Separating Control and Data Planes (2)



- Various efforts to separate the data and control planes began to emerge.

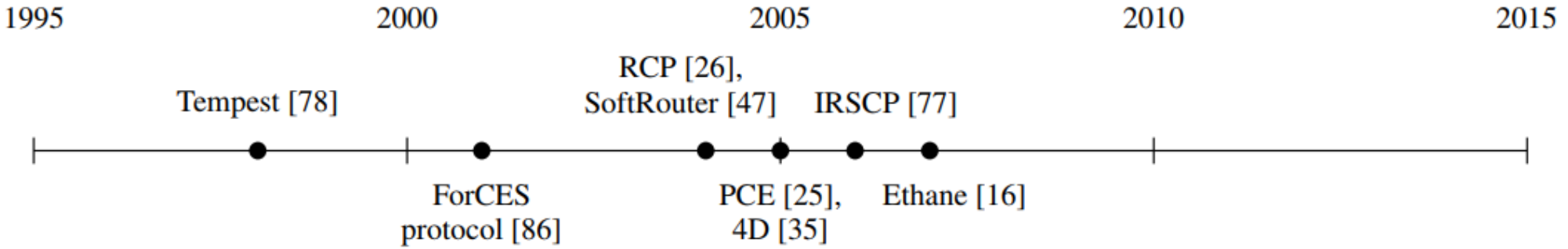
- Two important innovations:
  - 1- An open interface between the control and data planes
  - 2- Logically centralized control of the network

Separating Control and Data Planes (3)



- ForCES [86] (Forwarding and Control Element Separation) proposed a standard, open interface to the data plane to enable innovation in control-plane software.
- SoftRouter [47], IRSCP [77], RCP [26], 4D [35]: Logically centralized routing protocols.
- 4D: four main layers: data plane (processing packets based on configuration rules), discovery plane (collecting topology and traffic measurements), dissemination plane (installing packet processing rules), decision plane (logically centralized controller)

Separating Control and Data Planes (4)



- Several groups proceeded to design and build systems that applied this high-level approach to new application areas beyond route control.
- The Ethane project [16]: logically centralized, flow level solution for access control in enterprise networks. Ethane reduces the switches to flow tables that are populated by the controller based on high-level security policies. The simple switch design in ethane became the basis of the original OpenFlow API.

OpenFlow and Network OSes

- Generalizing network devices and functions
  - OpenFlow [51]: The OpenFlow API and OpenFlow switch
  - NOX [37]: A Controller platform (network OS) that enabled the creation of many new control applications.
- Distributed state management techniques
  - Running multiple controllers is crucial for scalability, reliability, and performance, yet these replicas should work together to act like a single, logically centralized controller.
  - Onix [46], ONOS [55]: Deployed distributed systems techniques to satisfy the state consistency and durability requirements.