

ECSE-6600: Internet Protocols

Informal Quiz #02

Shivkumar Kalyanaraman:
GOOGLE: "Shiv RPI"
shivkuma@ecse.rpi.edu



Review of Networking Concepts (II): Informal Quiz

T F **Review of Networking Concepts (II)**

- Without layering, you would need to translate between all possible combinations of applications and networking systems functions
- The interface between layers seldom outlives the technology used to implement each layer
- The service interface defines the message formats and conventions exchanged with the peer.
- A layer can use the services provided by the layer just above it in the layering stack
- One key difference between the Internet and the Telephone network is that the latter (I.e. telephone network) places its intelligence at the end-systems
- The end-to-end argument strongly encourages placement of the functions at the highest layer where it can be completely and correctly implemented.
- The end-to-end argument precludes function-placement at lower layers at all costs, and does not even allow it for performance gains
- The end-to-end argument is a principle, I.e. it can never be violated in Internet design

T F **Review of Networking Concepts (II)**

- Network architecture should specify both the placement of functions and detailed guidelines on how to best implement them
- The application-layer framing suggests that it would be advantageous to have application visibility and control mechanisms into as many networking functions as possible
- RTP is a protocol designed to fit the ALF philosophy, I.e. it provides an application-extensible transport layer format and functionalities
- Performance considerations are always absolute (I.e. relative performance questions such as: how much is A faster than B are not entertained)
- A useful rule of thumb from Amdahl's law is that we should focus our performance improvement efforts on the portion of the system that accounts for a significant share of performance (I.e. the "common case")
- Performance objectives and metrics depend upon the perspective taken (operator/user vs designer/implementor)
- A useful general rule of thumb in system design is to tradeoff scarce resources and optimize on abundant resources (I.e. be frugal about what is already abundant)