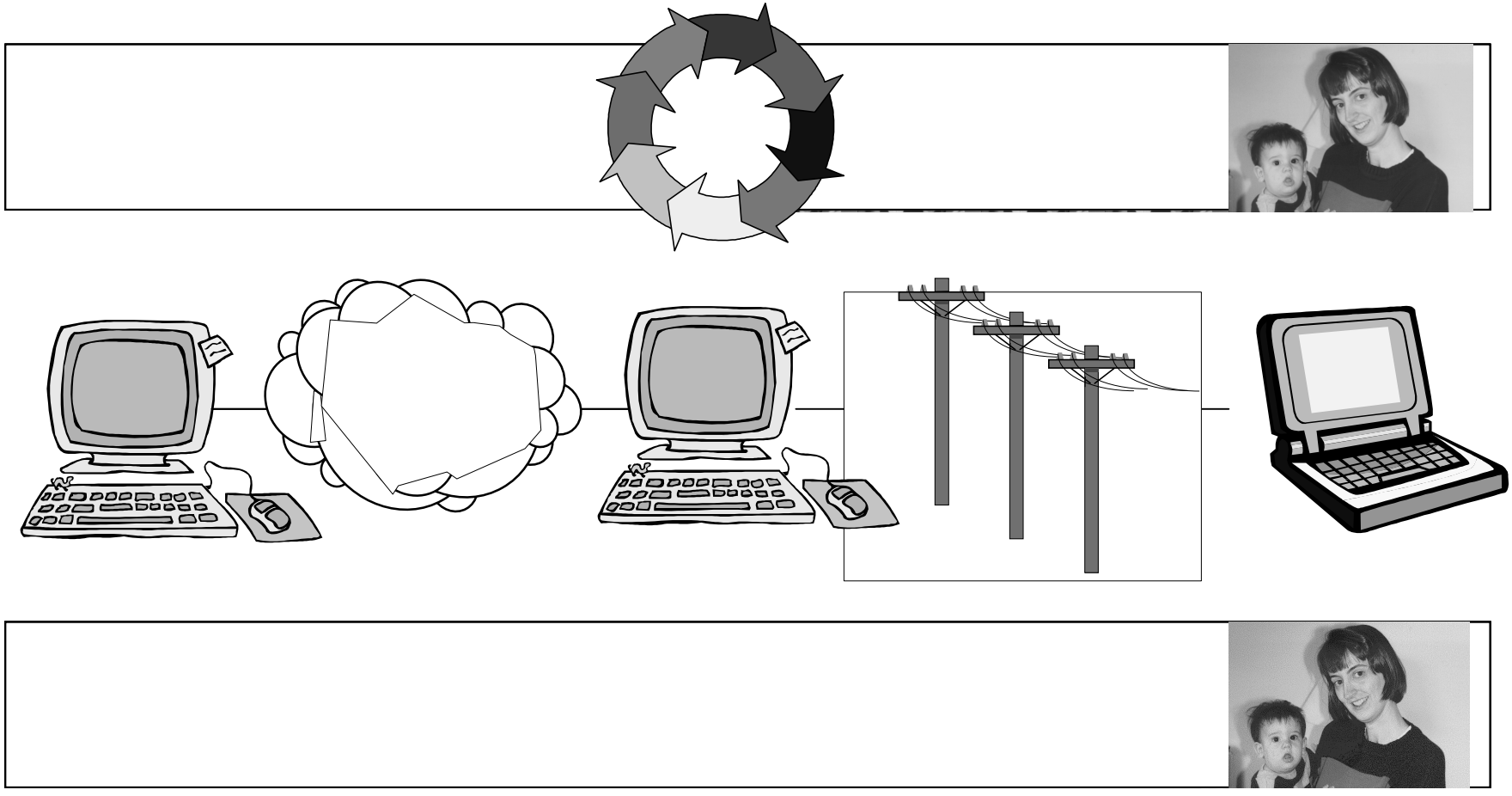
Conductor: A Framework for Distributed Adaptation

Mark Yarvis, Peter Reiher, and
Gerald J. Popek
University of California, Los Angeles

Hot Topics in Operating Systems March 29-30, 1999

Why adaptation?



A Need for Adaptability

- Increasing network complexity
  - Bandwidth, $ cost, power consumption, security, latency, jitter
  - Many dissimilar links
- Application writers assume a minimum level of network service
  - Leads to no useful service or high user costs

The Adaptive Solution

- Curb application's use of the network
  - Provides cost/benefit balance
- Types of adaptation
  - Distillation, compression, encryption, prioritization, pre-fetching, caching, buffering
  - Lossy or lossless
  - Data-type specific or general

Adapting with Conductor

- Adaptation is not new
  - Typical focus: last mile
- Conductor supports network complexity
  - Distribution and composition of adaptors
  - Dynamic configuration
  - Reliability despite arbitrary adaptations
  - Application transparency

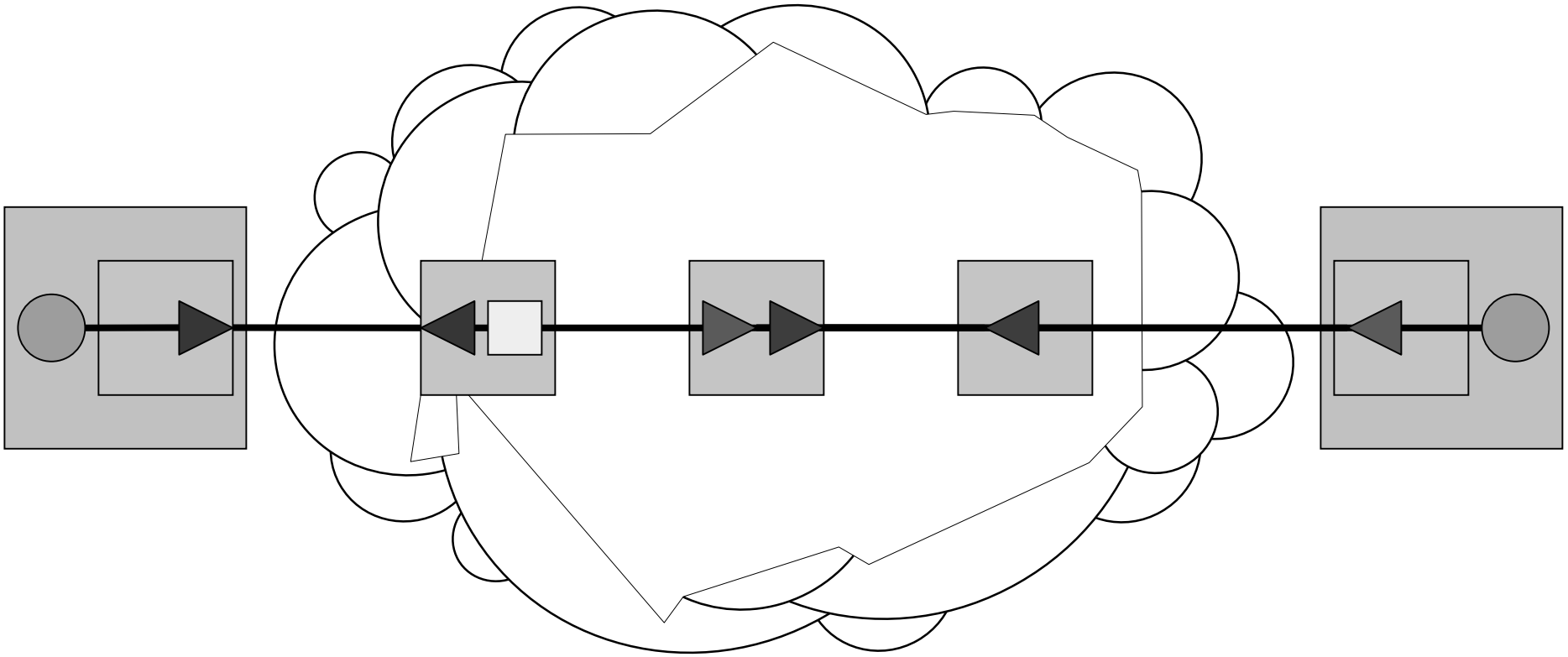
Roadmap to the rest of the talk ...

- Architecture
- Key Capabilities
- Related Work
- Development Status
- Conclusions

The Conductor Architecture

- The Conductor Framework
  - Deployed on network nodes
  - Provides interception at the stream level
  - Allows selection and deployment of adaptors
- Adaptors
  - Control content and timing of application-level protocols
  - Frequently paired
  - Composable

The Conductor Architecture



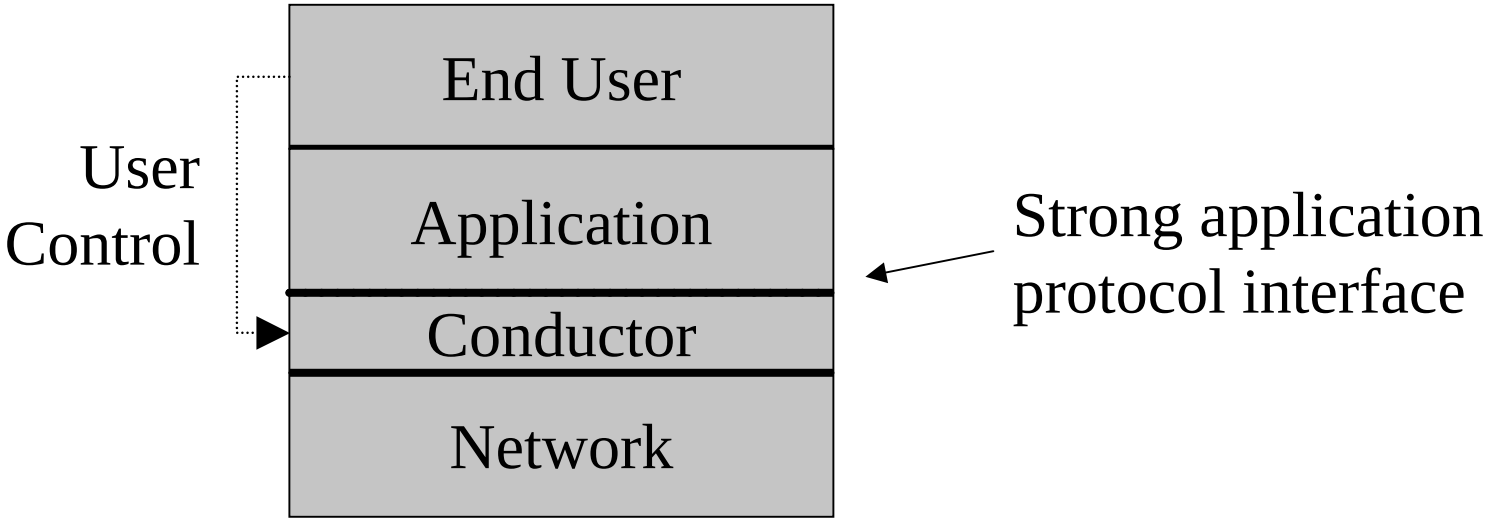
Key Capabilities of Conductor

- Application transparency
- Distributed adaptation
- Reliable transmission
- Dynamic configuration

Application Transparency

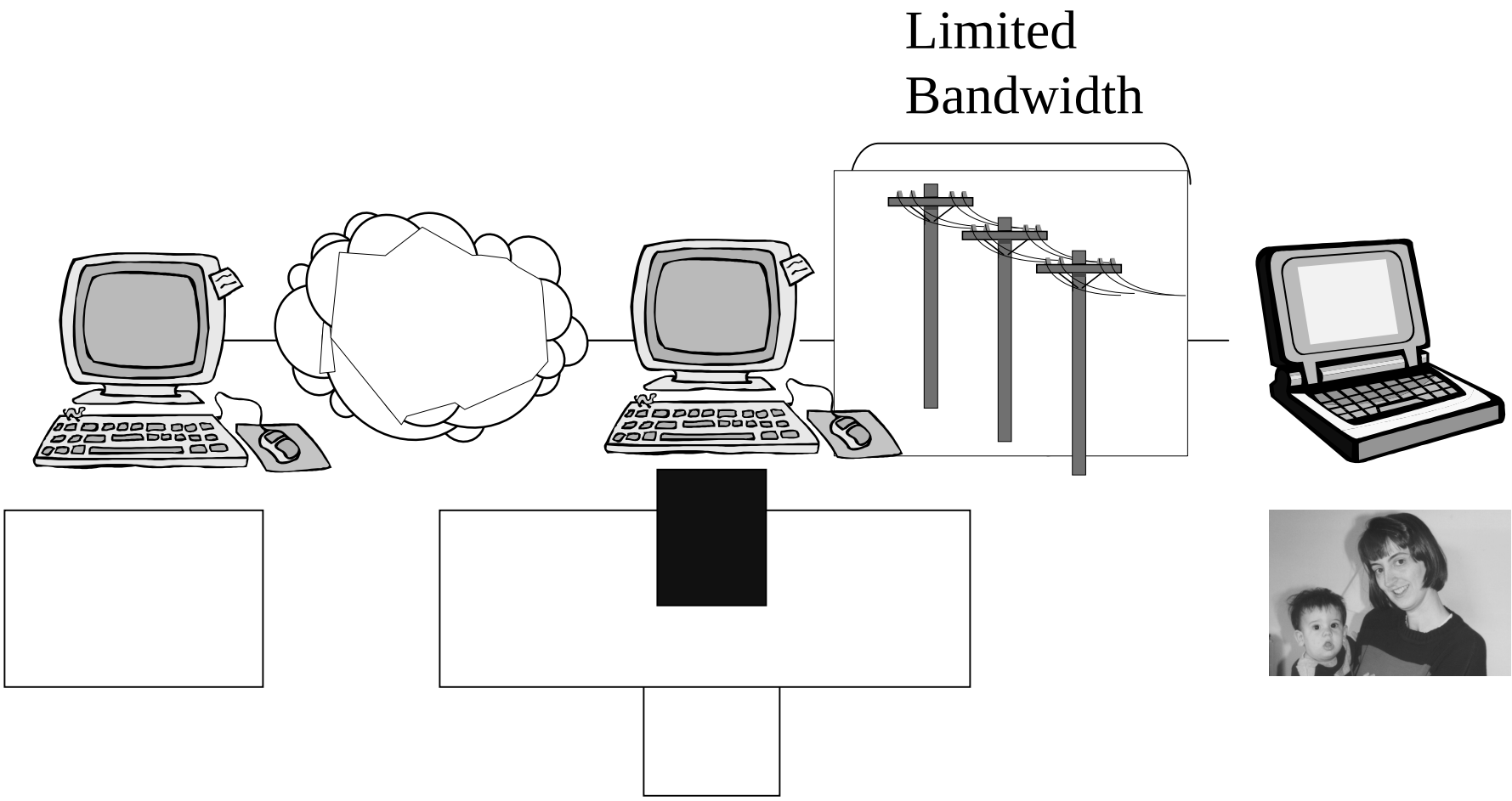
- Want to maintain application/adaptation separation
  - Free developers of complexity
  - Allow new or unforeseen adaptations
- But, aware applications are more adaptable
- Best of both worlds
  - Can extend a transparent system with an awareness API

Application Transparency

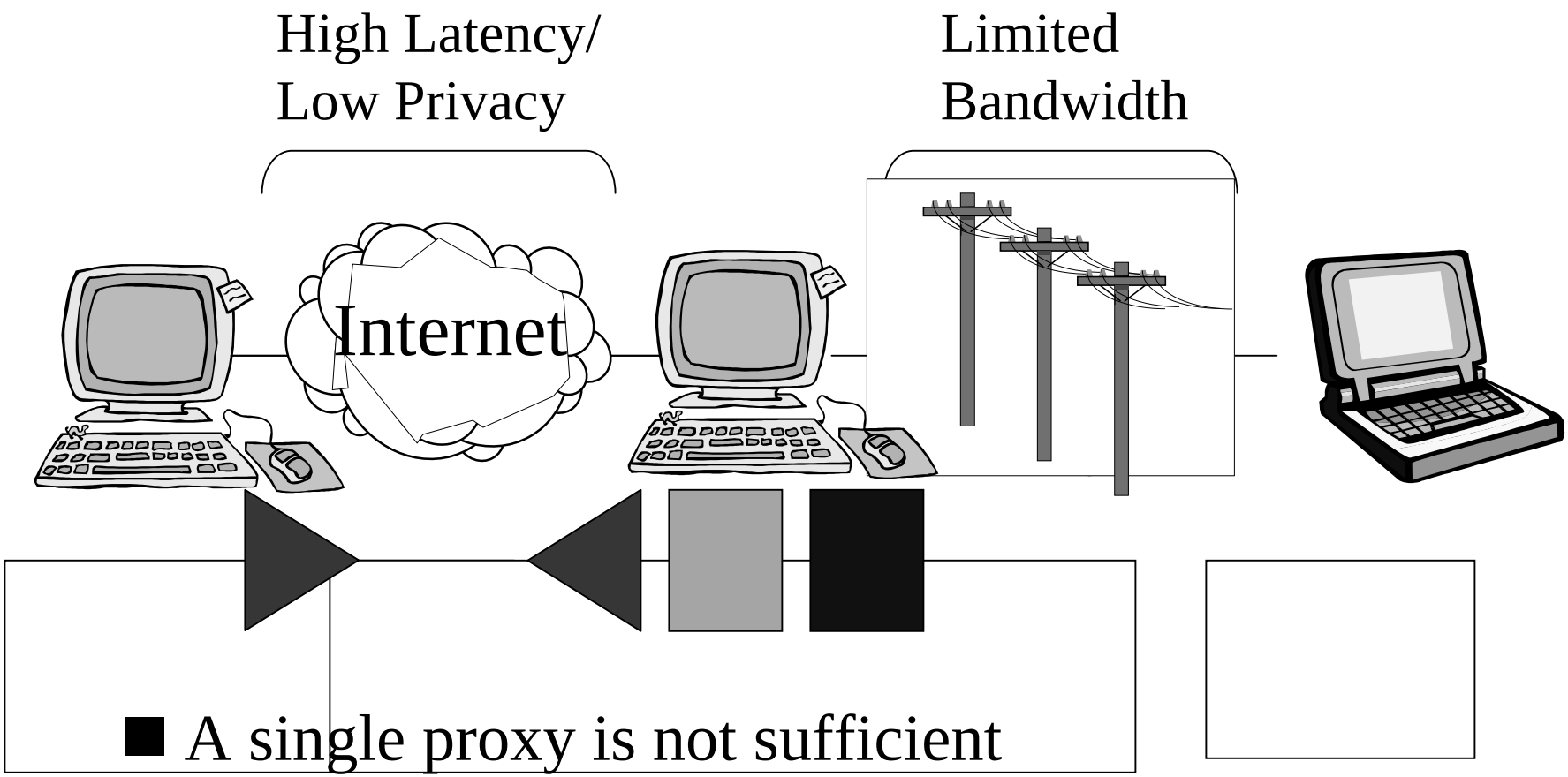


- Adaptation exploits flexibility in protocols
- Does not imply user transparency
  - E.g. video frame rate adaptation
- Requires user control

Distributed Adaptation



Distributed Adaptation



- A single proxy is not sufficient
- End-to-end adaptation is ineffective

Distributed Adaptation

- Distributed adaptation is required
- Conductor allows adaptation on any enabled node(s) along the data-path

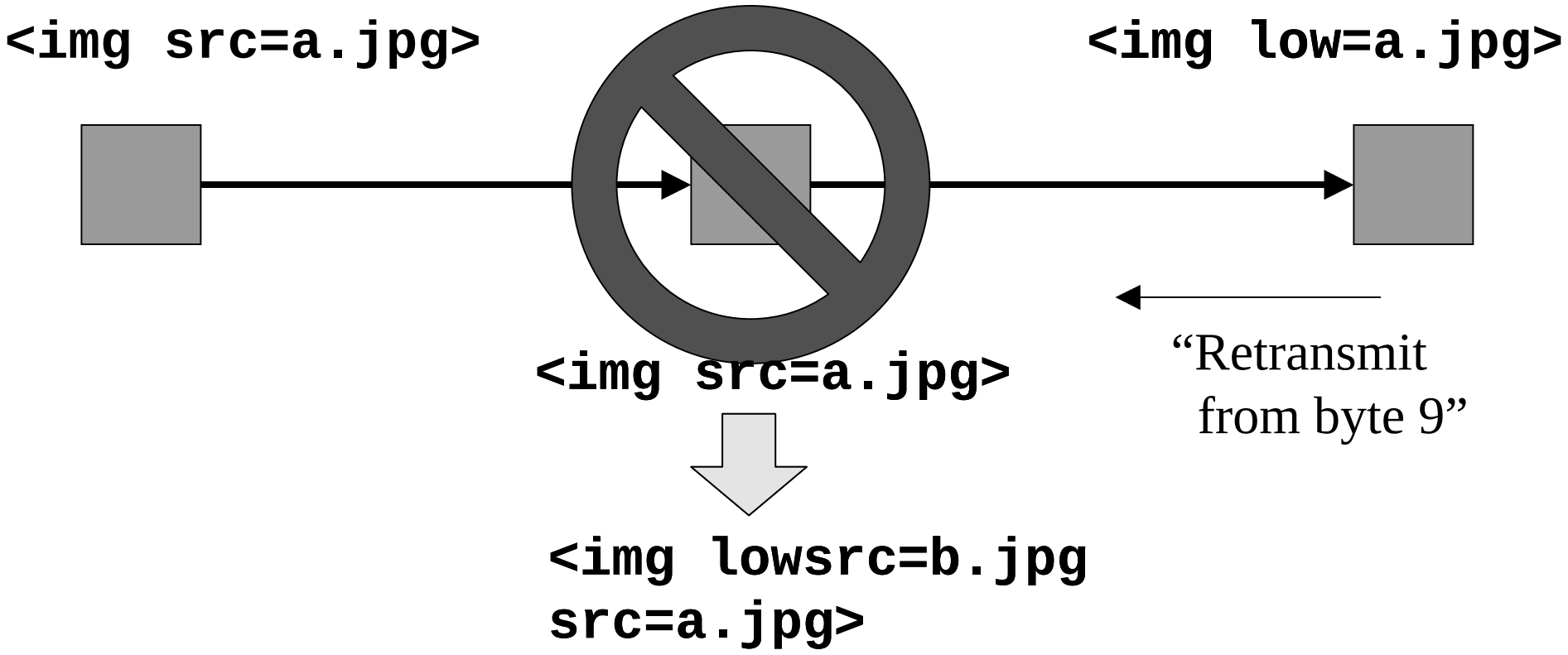
“A distributed system is one in which the failure of a computer you didn’t even know existed can render your own computer unusable”

-- Leslie Lamport, May 1987

Reliable Transmission

- Distribution introduces new points of failure
- End-to-end reliability typically assumes data immutability
  - retransmission by byte or packet count
- Adaptation modifies data in transit
  - need a new unit of retransmission

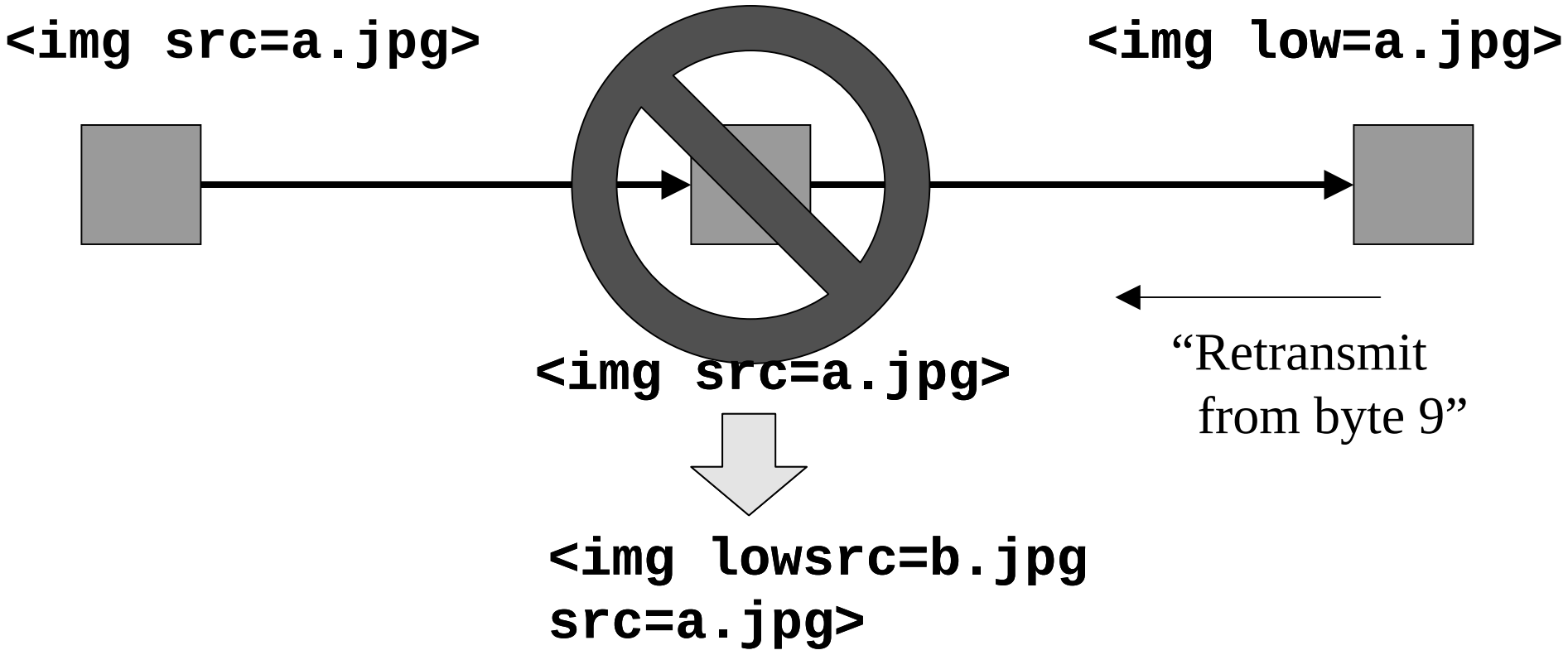
Reliable Transmission



Reliable Transmission

- Semantic Segmentation: a semantically meaningful unit of retransmission
  - segments defined by data type and adaptors
  - segment’s semantic meaning is preserved

Reliable Transmission

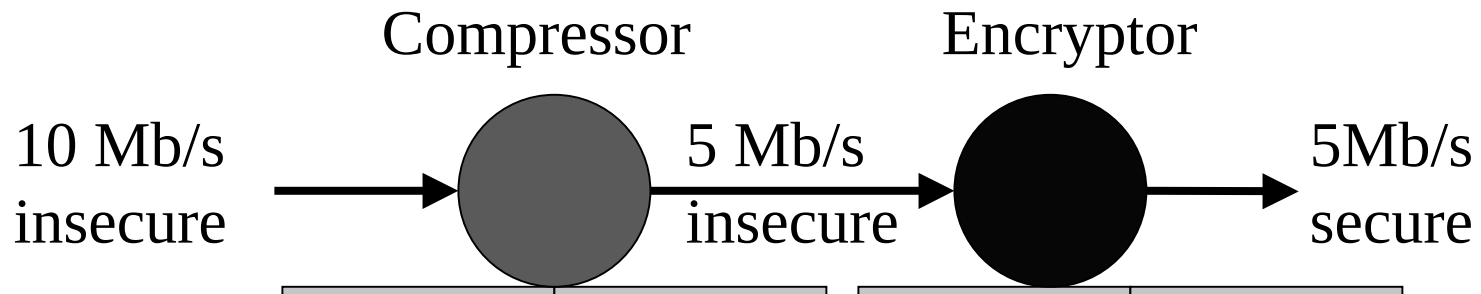


Dynamic Configuration

- Key goals
  - Select adaptors that solve the problem
  - Select adaptors that are compatible
- Conductor planning
  - Semi-centralized planning
  - Gather user preferences, link and node characteristics, and list of adaptors at single node
  - One round trip required

Dynamic Configuration

- Adaptors are self-descriptive
  - List of parameters
  - Input range and output function



	Compressor		Encryptor	
	In	Out	In	Out
Data Volume	any	* 0.5	any	* 1
Security level	any	* 1	any	5
Compressibility	> 30%	0%	any	0%

Development Status

- Current Status
  - Transparent connection interception
  - Planning mechanisms and simplified plan formulation
  - Adaptor instantiation and execution
- Future Work
  - Improved plan formulation
  - Reliability implementation
  - Application-aware API

Related Work

- Application-oriented adaptation
  - Odyssey (CMU)
  - Rover Toolkit(MIT)
- Proxy-oriented adaptation
  - TranSend, TACC (Berkeley)
- Protocol-oriented adaptation
  - Protocol Boosters (Bellcore, U. Penn.)
  - Transformer Tunnels (Rutgers)

Conclusions

- Adaptation allows users to control an application's use of network resources
- Complex networks require distributed adaptation
- Conductor provides arbitrary, user-controlled, application-transparent adaptation without compromising reliability