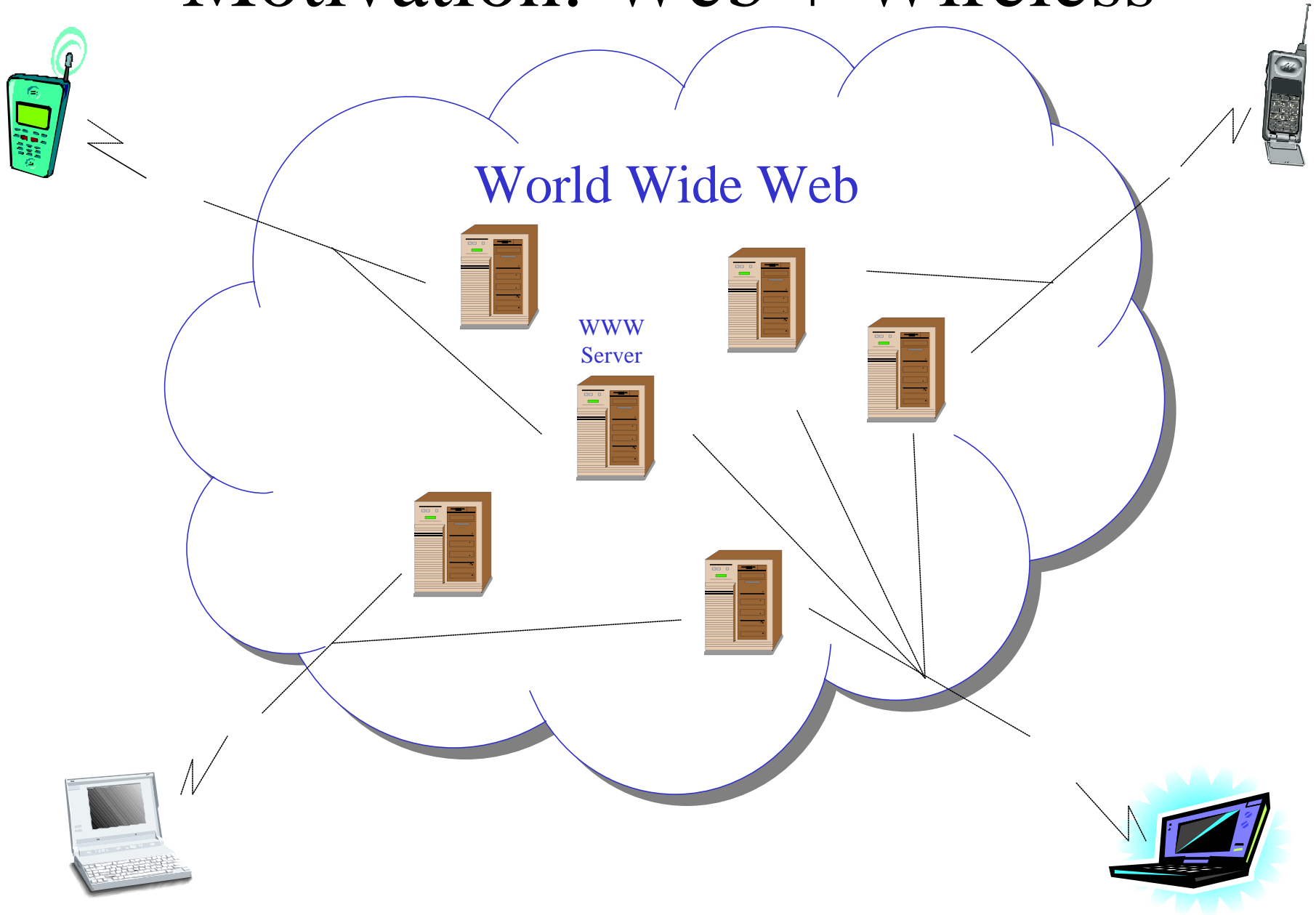# A Flexible Architecture for Customizing Web Streams for Wireless Clients

Jesse Steinberg and Joseph Pasquale
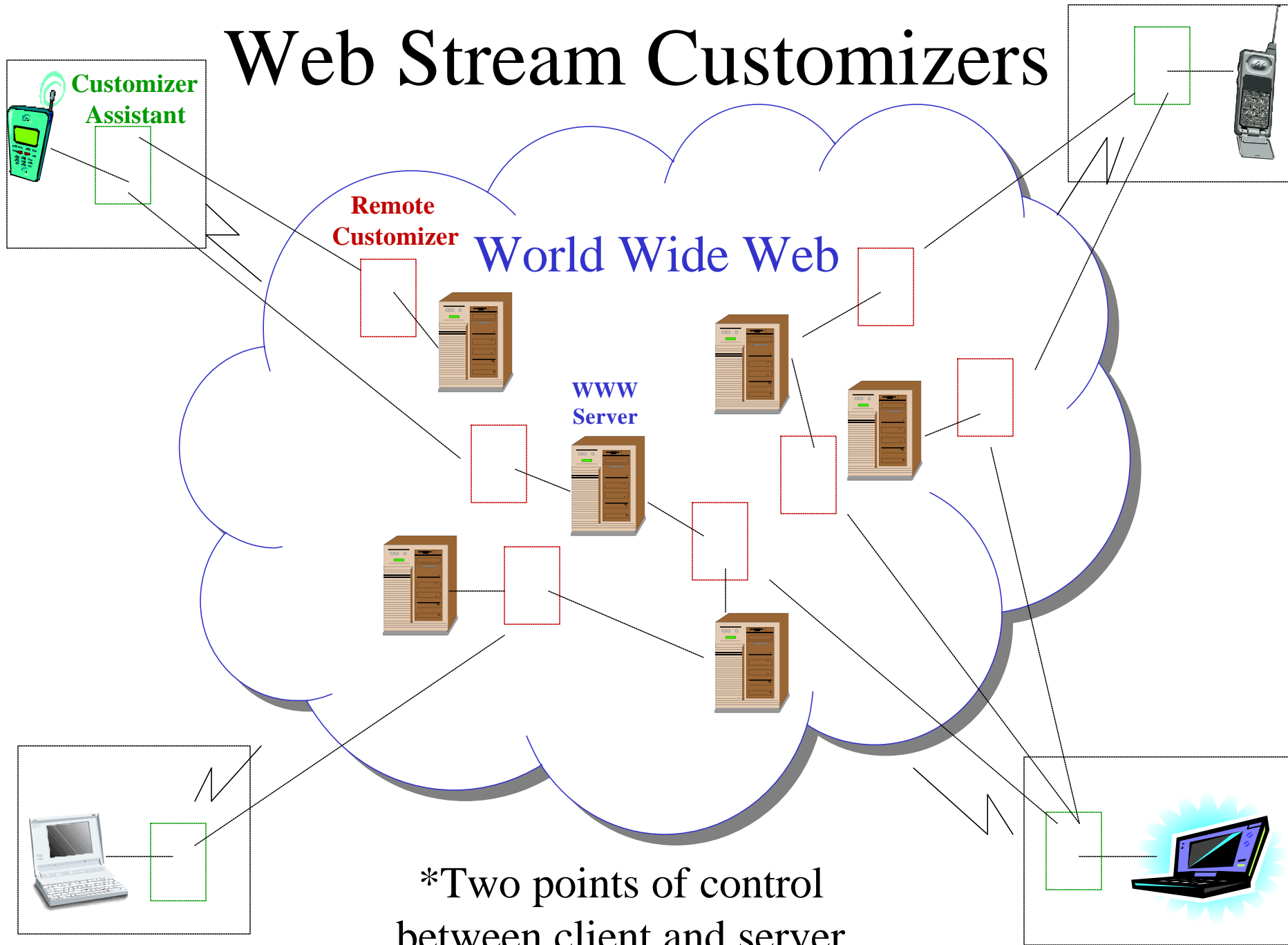
University of California, San Diego

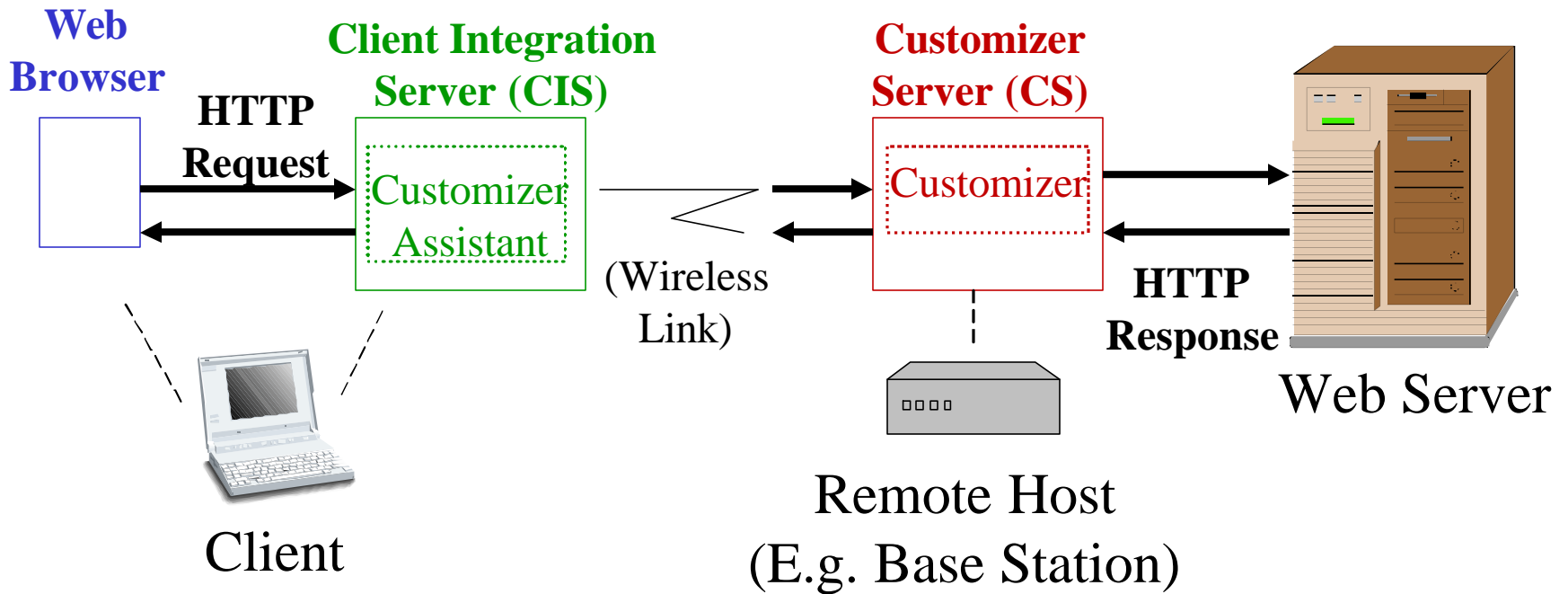Department of Computer Science and Engineering

Motivation: Web + Wireless

World Wide Web

WWW
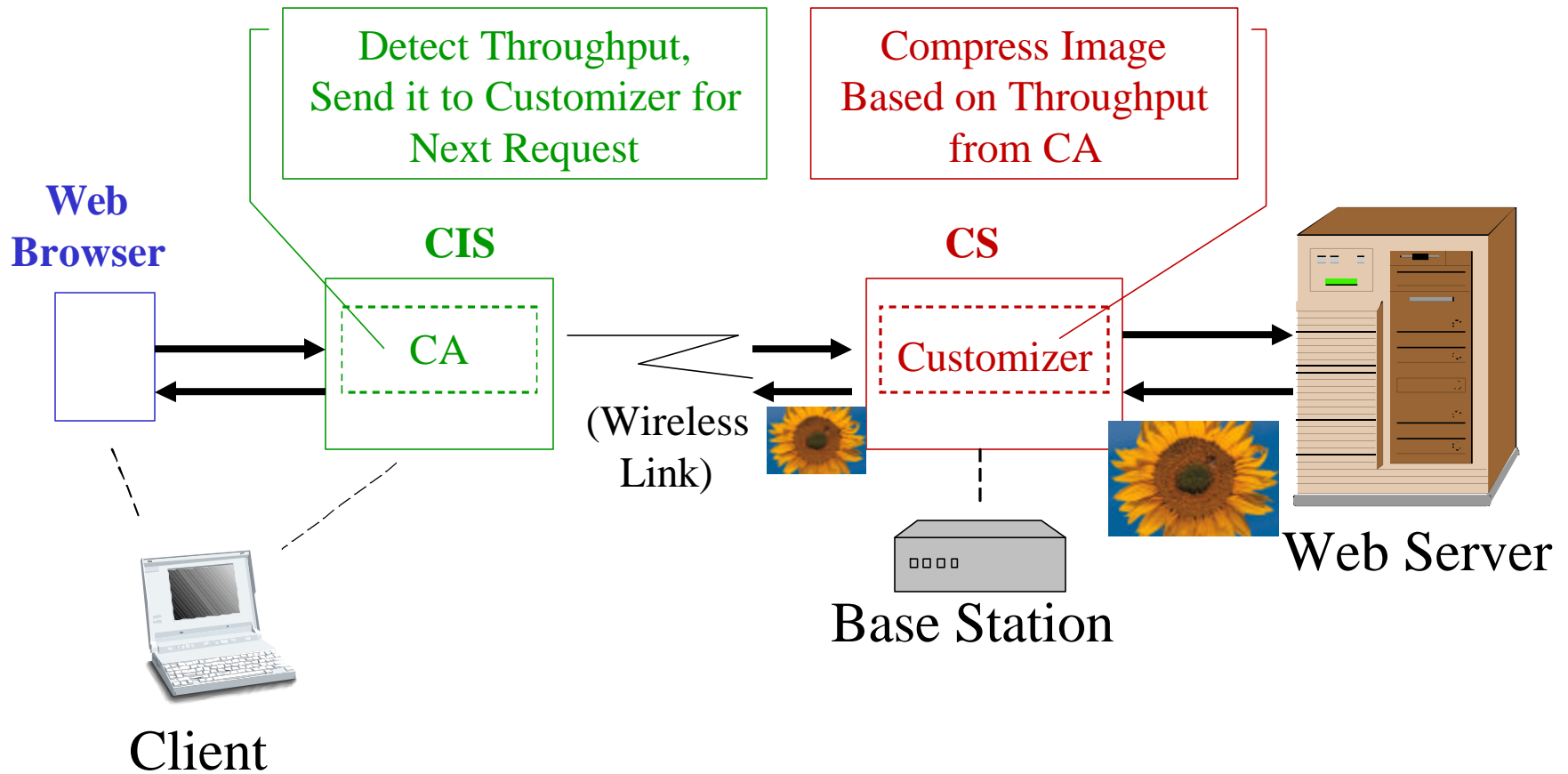Server

Web Stream Customizers

Customizer Assistant

Remote Customizer

World Wide Web

WWW Server

*Two points of control
between client and server

Communication Path

Web Browser — **HTTP Request** — **Client Integration Server (CIS)** — Customizer Assistant — (Wireless Link) — **Customizer Server (CS)** — Customizer — **HTTP Response** — Web Server

Client

Remote Host (E.g. Base Station)

*Customizer and Customizer Assistant can be dynamically deployed

Example: Adaptive Image Filter

Web Browser

Detect Throughput, Send it to Customizer for Next Request

Compress Image Based on Throughput from CA

CIS

CA

CS

Customizer

(Wireless Link)

Web Server

Client

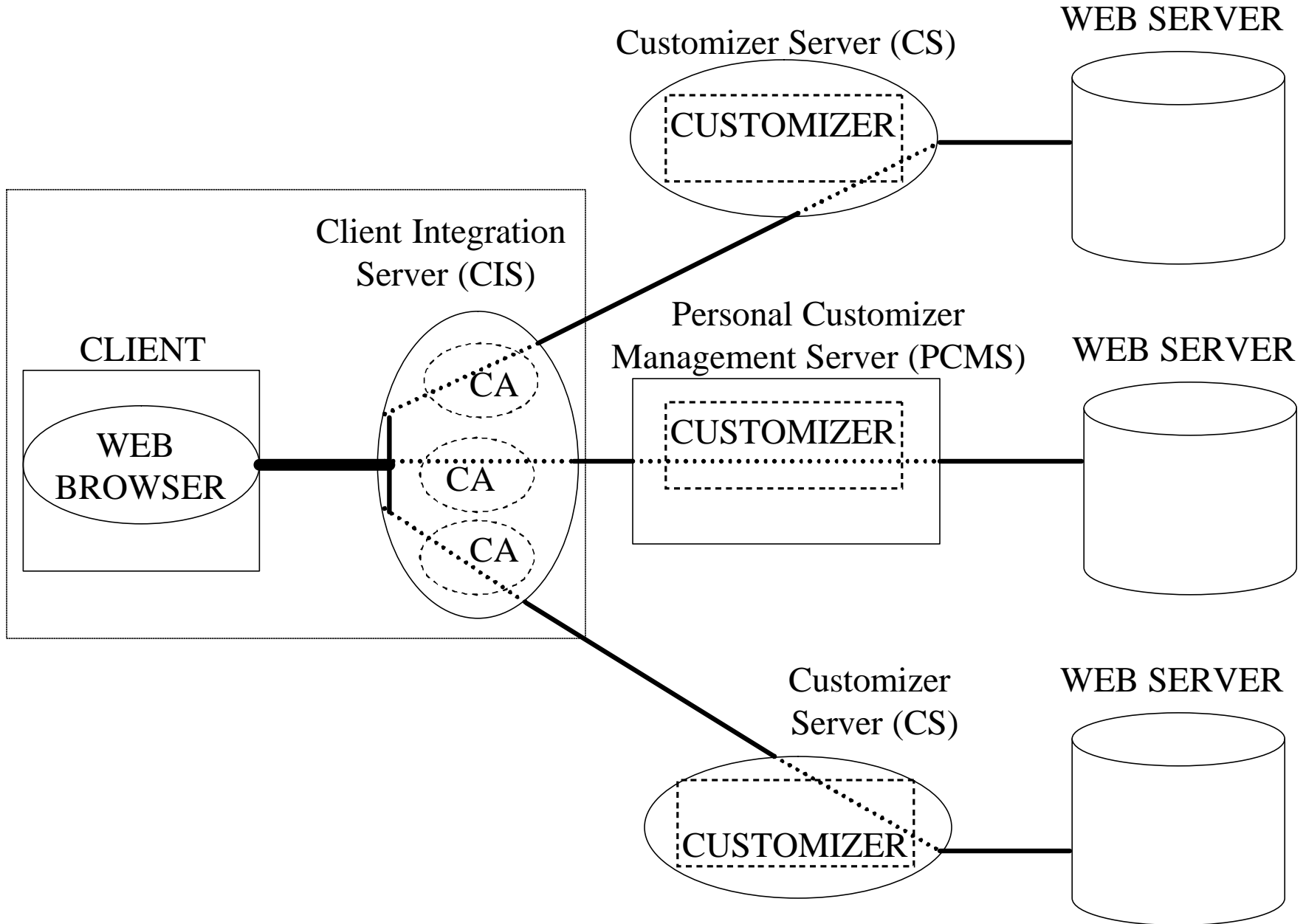Base Station

Where Do Customizers Run?

- Can be a third party server
    - Flexibility of location
- A personal server can be used
    - Personal Customizer Management Server (PCMS)
    - Take advantage of availability of user owned host or account
    - Can use resources such as persistent storage

Multiple Active Customizers

WEB SERVER

Customizer Server (CS)

CUSTOMIZER

Client Integration
Server (CIS)

Personal Customizer
Management Server (PCMS)

WEB SERVER

CLIENT

WEB
BROWSER

CA

CA

CA

CUSTOMIZER

WEB SERVER

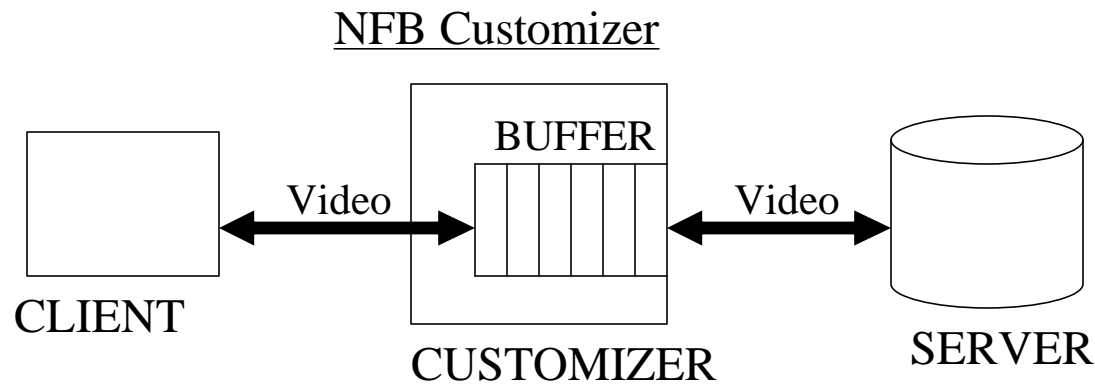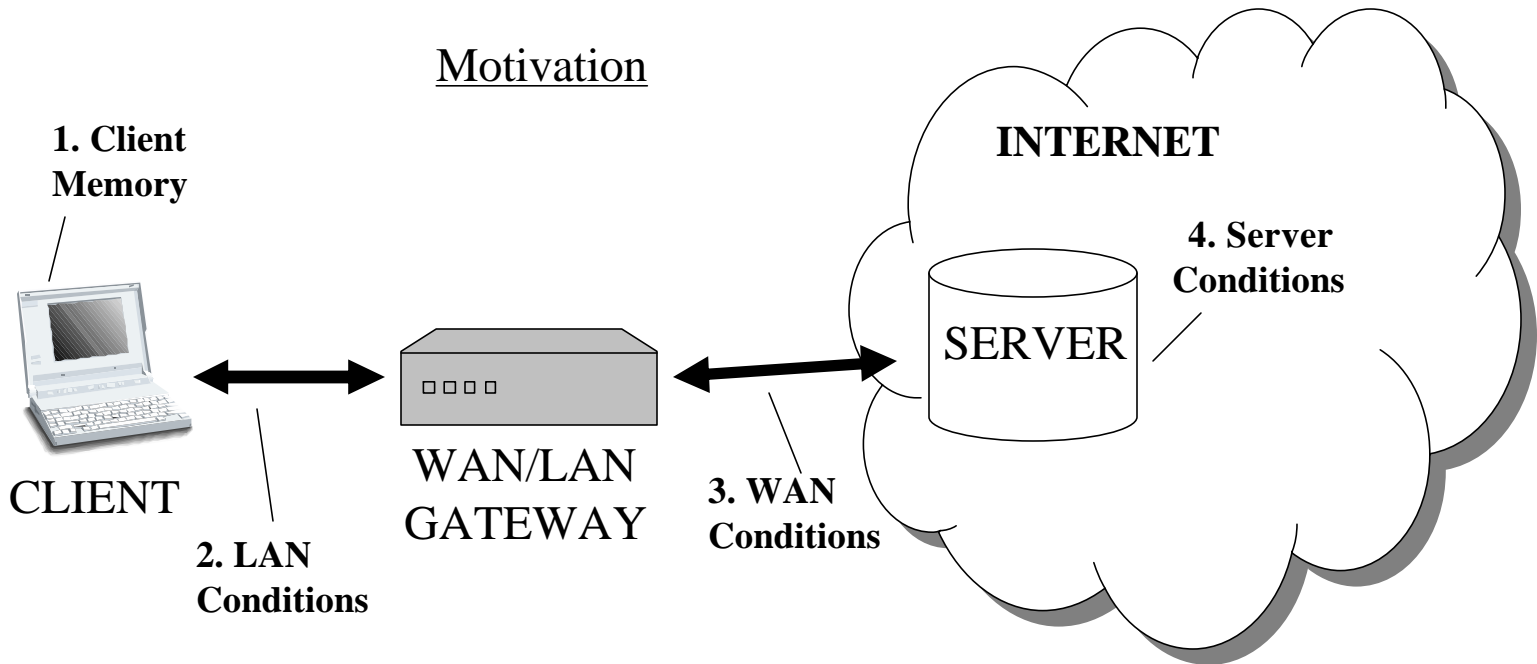Customizer
Server (CS)

CUSTOMIZER

Customizer Applications

- Adaptive Compression
  - Text, image filtering
- Transaction Reliability
  - Mask failures, store results at CS
- Selective Encryption
- Network Flow Buffering
  - Buffer and regulate streaming traffic
  - E.g. Streaming multimedia

Network Flow Buffer: Closer Look

Motivation

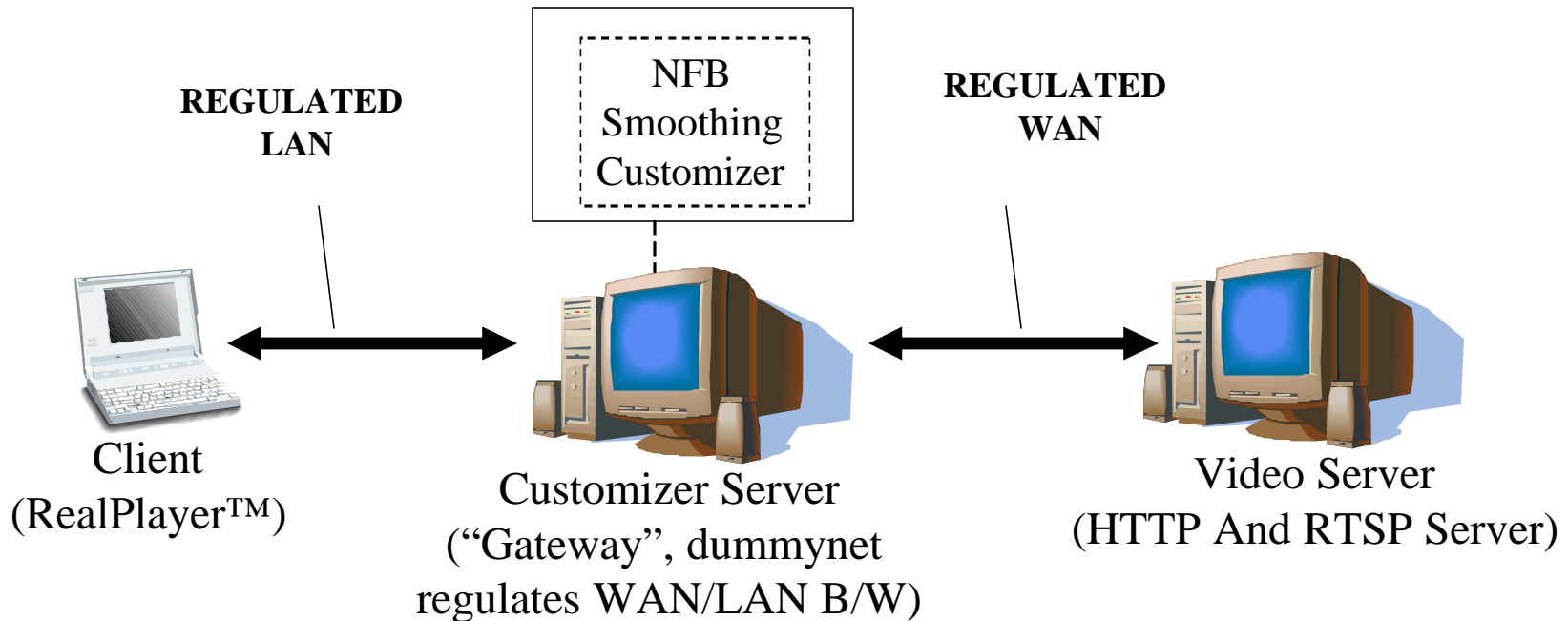**INTERNET**

1. Client Memory

CLIENT

WAN/LAN GATEWAY

2. LAN Conditions

3. WAN Conditions

SERVER

4. Server Conditions

NFB Customizer

CLIENT

Video

BUFFER

CUSTOMIZER

Video

SERVER

NFB Smoothing

Case 1:

Higher WAN
B/W
→

Lower LAN
B/W
→

Buffer Fills Over Time

Case 2:

Lower WAN
B/W
→

Higher LAN
B/W
→

Buffer Drains Over Time

* Goal: Maintain Smooth, Uninterrupted Video Playback

Smoothing Performance Evaluation

REGULATED LAN

NFB
Smoothing
Customizer

REGULATED WAN

Client
(RealPlayer™)

Customizer Server
("Gateway", dummynet
regulates WAN/LAN B/W)

Video Server
(HTTP And RTSP Server)
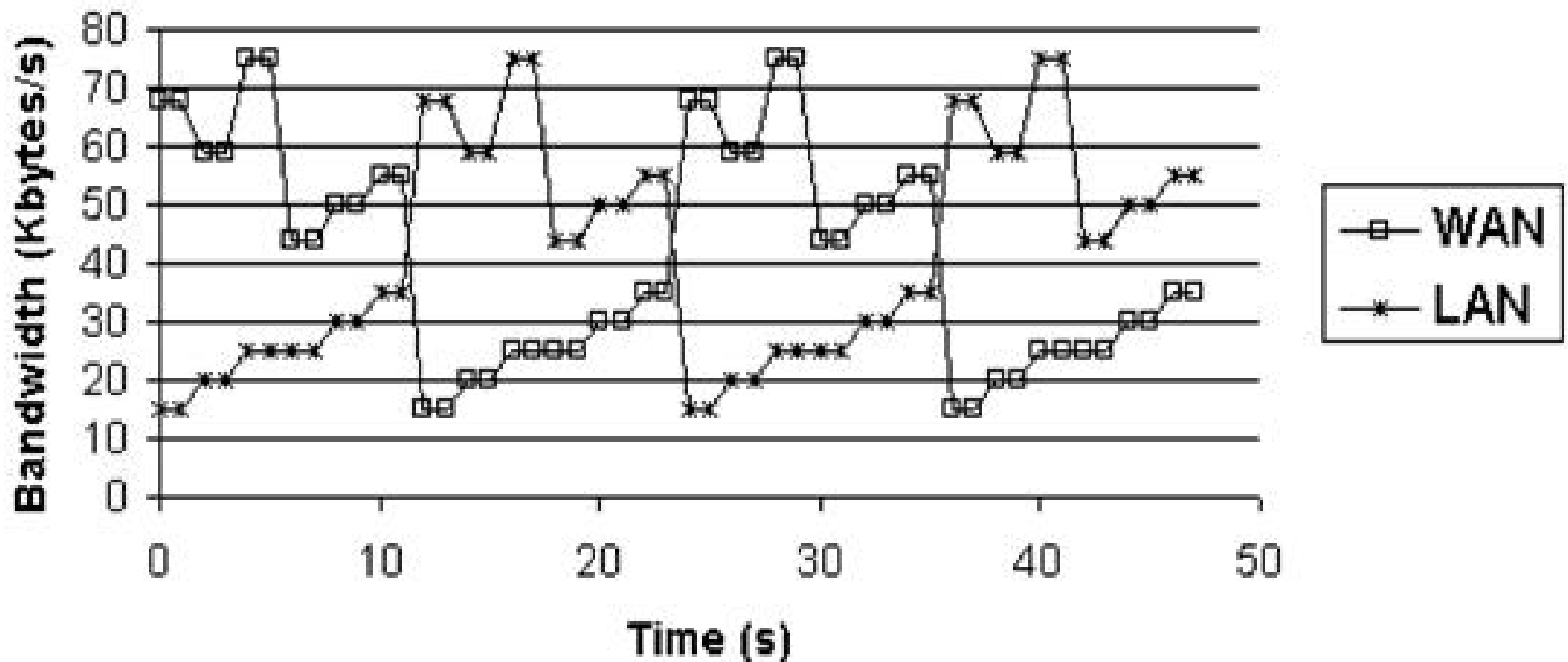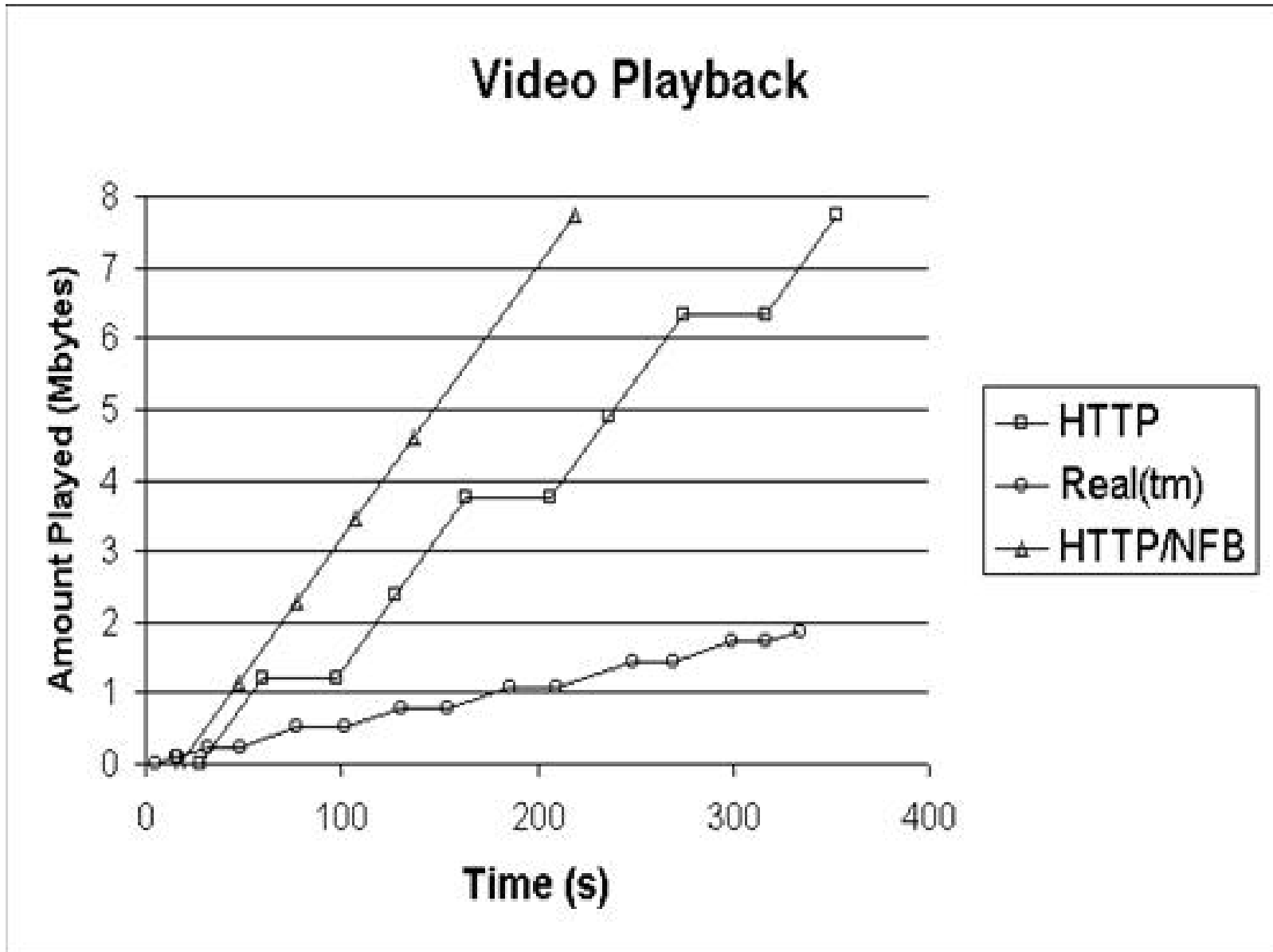
Smoothing Experiment

Bandwidth Cycle
2 cycles of 24 seconds each
12 bandwidth changes per cycle

Smoothing Results: Playback

Video Playback
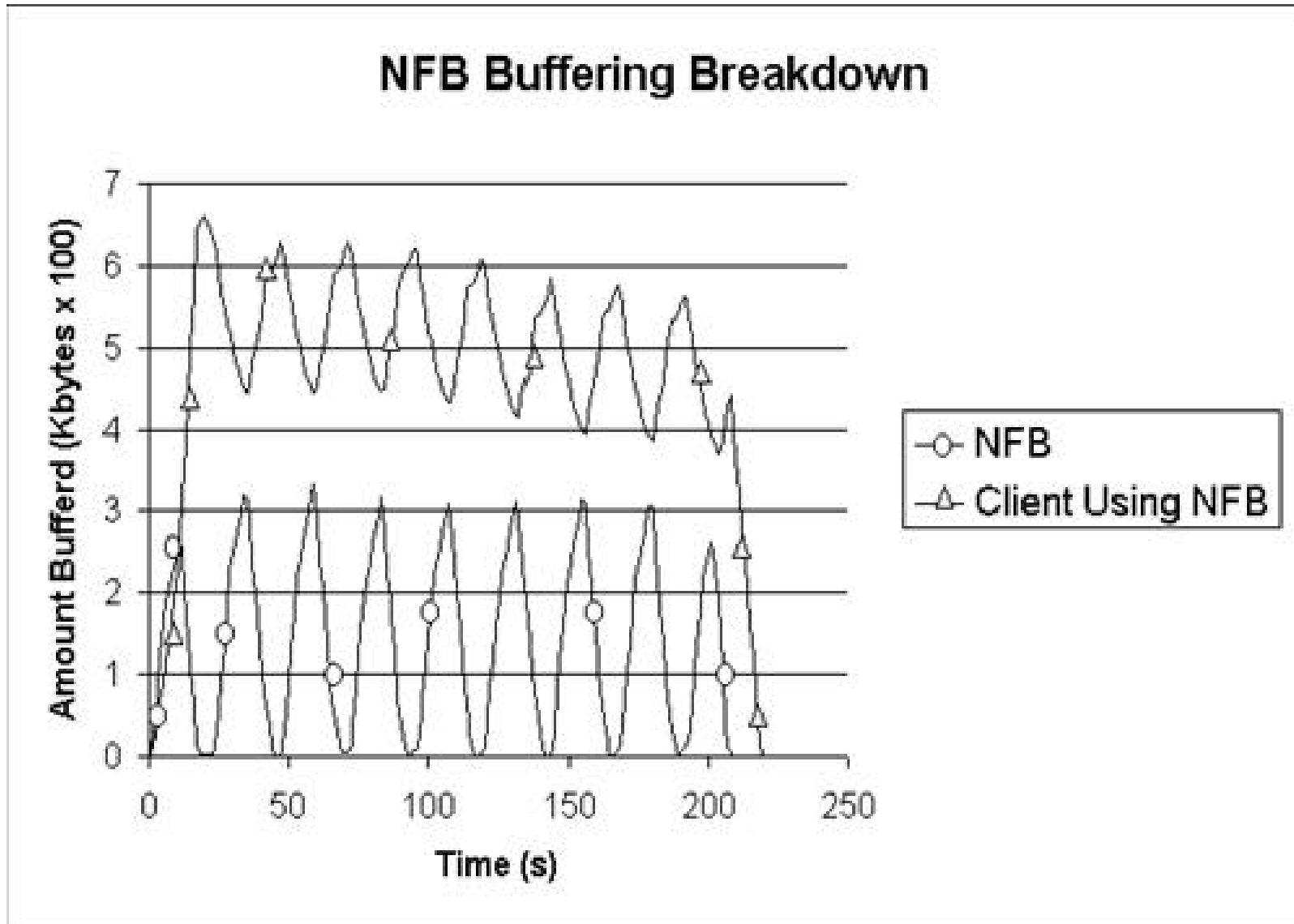
Smoothing Results: Buffering

Video Buffering

Legend:
- HTTP
- Real (tm)
- HTTP/NFB

X-axis: Time (s)
Y-axis: Amount Buffered (Kbytes x 100)

NFB Buffering Breakdown

Summary

- Novel Web middleware architecture for improved wireless web access
  - Remote computation, dynamic deployment, two points of control, callback programming model
- Supports a variety of applications
  - Filtering, encryption, transaction recorder, video buffering
  - NFB smoothing can improve video playback
- Implementation
  - Java-based and uses existing Web mechanisms

Customizers Are Efficient

- Customizer overhead ~ 4.8 ms
  - Roughly 1-5% of typical transfer times
- Typical transfer times from UCSD:
  - www.yahoo.com ~ 128 ms
  - www.suntimes.com/index ~ 404 ms
  - www.cnn.com ~ 475 ms
- Above doesn't consider performance improvements of the Customizer