DBAS: A Deployable Bandwidth Aggregation System

Karim Habak†, Moustafa Youssef†, and Khaled A. Harras‡
†Egypt-Japan University of Sc. and Tech. (E-JUST)
‡Carnegie Mellon University in Qatar

IFIP NTMS‘12
Motivation

- The proliferation of multi-homed or multi-interface enabled devices
Motivation

• The proliferation of multi-homed or multi-interface enabled devices
• These interfaces are utilized one at a time
Motivation

• The proliferation of multi-homed or multi-interface enabled devices
• These interfaces are utilized one at a time
• Exponential increase in mobile data demand
Bandwidth Aggregation Systems

• Availability of many solution for utilizing the available multiple interfaces
• The available solutions face a steep deployment barrier
  – Why ??
Solutions Shortcomings

• Requiring updating the legacy application
  – Ex: Application has to quantify its requirements
• Requiring updating the network infrastructure and introduce new components
  – Ex: Proxies and routers
• Requiring updating the legacy servers
• Introduce Kernel level modifications to the end nodes
  – Requires recompiling the kernel
Design Goals

- **Goal 1:** Avoid modifying the legacy applications while making use of their characteristics
- **Goal 2:** Avoid network infrastructure updates
- **Goal 3:** Avoid legacy servers updates while making use of this opportunity if available
- **Goal 4:** Avoid kernel level modifications to the clients
- **Goal 5:** Utilize the available interfaces to their maximum
- **Goal 5:** Capture the user preferences
System Architecture

- End-to-End communication
  - No infrastructure updates
System Architecture

• Application Characteristics Estimator
  – Qualitative measurements
  – Quantitative measurements
System Architecture

- Mode detection module
  - A server listening on specific port
  - Specifies either the server DBAS-enabled or not
System Architecture

- Interface Characteristics Estimator
  - Estimates the available bandwidth at each interface
  - Uses destination based estimates in case of DBAS-Enabled servers
System Architecture

- **Scheduler**
  - Schedules the packets and/or the connections on the different network interfaces
System Architecture

• Received Data Reordering Module
  – Used only in packet oriented mode
  – Reorder the packets before giving them to the application
System Architecture

• User Interface Module
  – Obtains user’s preferences and interface usage policies
Scheduling Techniques

• Only-One:
  – Reflects the current OS’s scheduling technique which selects only one of the available network interfaces

• CO Round Robin
  – Assigns the connections to the different network interfaces in a rotating basis
  – Does not take into account the interfaces characteristics and applications characteristics
Scheduling Techniques

• CO Weighted Round Robin
  – Assigns the connections to the different network interfaces in a rotating basis weighted by the interface estimated bandwidth such that higher bandwidth interfaces get assigned more connections
  – Takes the interfaces characteristics into account while not taking the applications characteristics
Scheduling Techniques

• CO Maximum Throughput
  – Assigns the connections to the different network interfaces such that it minimizes the time need to finish the current system load in addition to the new connection’s load
  – Takes both the interfaces characteristics and the application characteristics into account
Scheduling Techniques

• PO Round Robin
  – Assigns the packets/chunks to the different network interfaces in a rotating basis
  – Does not take into account the interfaces characteristics and applications characteristics

• CO Weighted Round Robin
  – Assigns the packets/chunks to the different network interfaces in a rotating basis weighted by the interface estimated bandwidth
  – Takes the interfaces characteristics into account while not taking the applications characteristics
Implementation

• DBAS Middleware
  – It is implemented as a Layered Service Provider (LSP)
  – It is installed as a part of the TCP/IP stack in Windows OS
  – It intercepts socket-based connection requests and assign proper network interfaces to them or distribute their data across the different interfaces
Implementation

• DBAS Monitoring Application
  – It is used to captures the user preferences and interfaces’ usage policies
  – It is also used to monitor DBAS middleware and its estimates
Evaluation

• Network Topology
  – Client node equipped with multiple network interfaces (IF1 and IF2)
  – Server node
  – Intermediate node with NIST-NET network emulator is used to change the network interfaces characteristics
Evaluation

• Applications Models
  – Small load application represents Internet browsing
    • Generate connections with length follows exponential distribution with mean 22.38KB
    • Generate connections following a Poisson process with mean 13 connection/sec
  – Large load application represents P2P and FTP
    • Generate connections with length follows exponential distribution with mean 285KB
    • Generate connections following a Poisson process with mean 1 connection/sec
Results (Interfaces Heterogeneity)

Using interfaces characteristics estimates achieves up to 396% throughput gains
Without using interfaces characteristics scheduling may achieve lower performance than single interface even in PO scheduling
Results (Applications Heterogeneity)

Using applications characteristics estimates achieves up to 130% throughput gains.
PO Scheduling is not affected by applications characteristics.