

HTTP Enhancements for Hybrid / Satellite Networks

A. Bhalekar / J. Baras

BACKGROUND

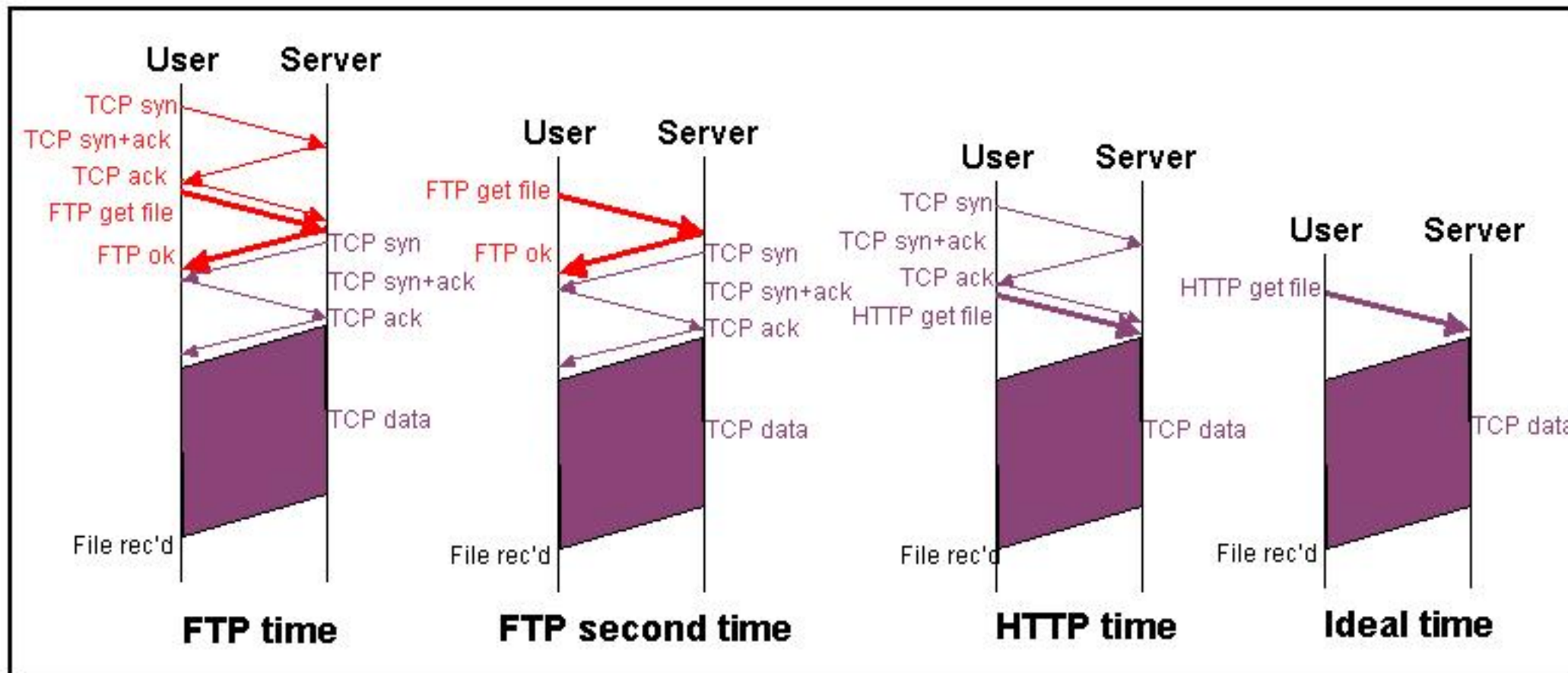


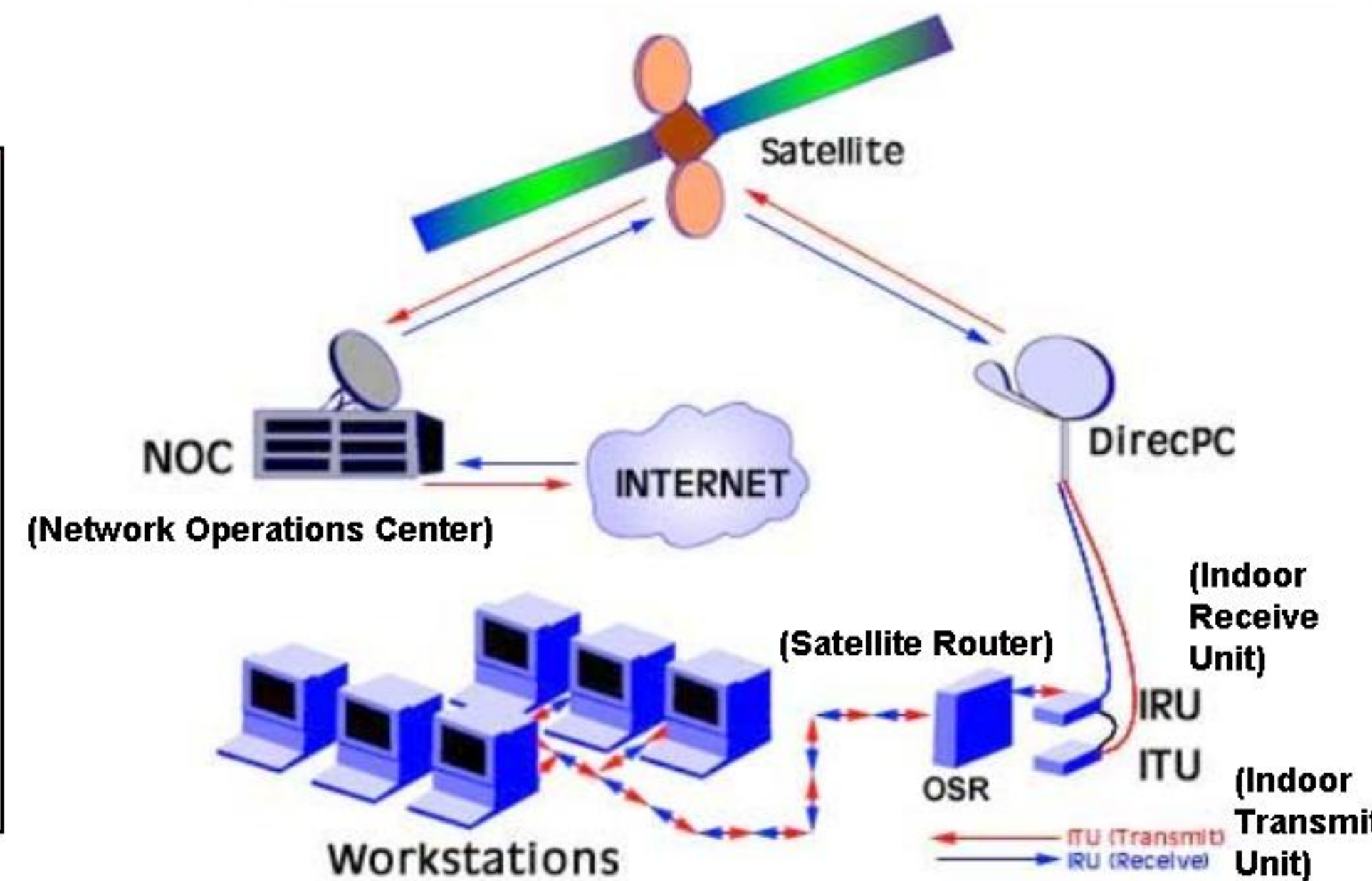
Figure I

Figure I

FTP – 1st packet transfer requires 3 RTT's (including: TCP SYN, ACK, GET, OK, ACK, DATA) + Data transfer time
 Subsequent packets need 2RTT's + Data transfer time
 HTTP – 1st packet needs 2 RTT's (TCP SYN, ACK, GET, DATA) + data transfer time and the subsequent need just 1 RTT + Data transfer time

Figure II

The basic block diagram of the Internet via Two-way Satellite System is as depicted. This "two-way" solution is the latest development and is also most convenient as it does not require a separate ISP for the outbound requests. Instead the requests and downloads are all via the satellite



Source: Hughes Network Systems

Figure II

PROMISING DEVELOPMENTS

Larger cwnd and Multiple Connections

MOTIVATION

Figure out a way to, either, toggle between the two, or have slightly larger cwnd's on each of the multiple connections

ADVANTAGES

A Larger congestion window will help reduce multiple round trips for a single HTTP request.
 A larger cwnd is ideal on a single connection, for large images (or elements) and moderate in number one one web-page
 Multiple connections are ideal for web-pages with lot of smaller sized elements
 We could use, for example, JavaScript, to tell us in advance what kind of element number and size the web-page contains (meta-information)

DRAWBACKS

Implementation of this scheme can be quite unwieldy

Data Compression and Delta Encoding

MOTIVATION

Web-pages have potential for aggregation as they are composed of several files. The combination of the two methods gives huge savings in response time. A delta is the difference in the web-page, between the latest and previous request.

ADVANTAGES

Deltas can be further compressed
 Deltas are at LEAST less than half the original size of the web-page
 Deltas easy to implement due to library functions (e.g. Gzip)
 Deltas can be cached at the servers
 Deltas can be most obviously used by the "Push" applications

DRAWBACKS

Deltas useful only if page accessed multiple times
 To calculate deltas, the server has to cache various versions of updates of a page
 The usage of deltas depends upon whether the data is delta-eligible or not

CONCLUSIONS

- Deltas could be created and cached after the resource changes even before the request arrives, to save time! This can be done because statistics show that HTTP requests repeat over time. This is the most promising solution
- We could develop a new caching algorithm for the client, and try to incorporate it within HTTP
- We could fruitfully use meta-information to increment/decrement cwnd size and/or setup a certain pre-determined number of connections

FUTURE WORK

- Further investigate the best possible solution from the ones mentioned above
- Run detailed simulations of the HTTP enhancements
- Formalize results showing the savings in the metric considered (time)

HTTP ver 1.0

MOTIVATION

A better request-response protocol as compared to existing FTP

ADVANTAGES

1st packet transfer needs 2 RTTs (FTP needs 3RTTs) and the Subsequent transfers need just 1RTT (FTP needs 2RTTs)

DRAWBACKS

New connections for each request. Connection setup overhead for each request is high

HTTP ver 1.1

MOTIVATION

Eliminate setup overhead in 1.0

ADVANTAGES

Caching and Pipelining of multiple HTTP requests is incorporated

DRAWBACKS

Single HTTP responses may require multiple round trips as HTTP requests can be longer than TCP segment size

DEVELOPMENTS

P-HTTP

MOTIVATION

HTTP requests tend to repeat over time

ADVANTAGES

If previously setup connections persist and requests repeat, a considerable saving in access time is obtained

DRAWBACKS

Complexity at the application layer
 Due to segmentation and reassembly.
 11% Improvement due to slow web

Conditional Statements

MOTIVATION

A conditional alternative to the regular GET statement using cache validation

ADVANTAGES

90% of the data on the requested web page need not be resent using, for example, a command like GETLIST

DRAWBACKS

A multitude of options in GET statements lead to lack of an optimal algorithm. Cache-bursting may occur

Change Underlying Protocol

MOTIVATION

TCP isn't ideal for requests-responses
 Is there a transport layer protocol on which HTTP performs better?
 Can we emulate that behavior?
 Can we modify/replace TCP?

ADVANTAGES

T/TCP & S-TCB cache the state information of the protocol.
 Connection caching protocols are very useful even for single web-page hits.

DRAWBACKS

Difficult to implement

Change Web-content

MOTIVATION

Current web-content is not ideal for compression for our purpose

ADVANTAGES

Using CSS-Cascading Style Sheets and PNG-Portable Network Graphics (not JPG/GIF) and hence changing web content suitably will help save time

DRAWBACKS

A universal change is required, else results are not immediately gratifying