DWDM RAM

Data@LIGHTspeed

A Platform for Data Intensive Services Enabled by Next Generation Dynamic Optical Networks

D. B. Hoang, T. Lavian, S. Figueira, J. Mambretti, I. Monga, S. Naiksatam, H. Cohen, D. Cutrell, F. Travostino

Gesticulation by Franco Travostino

DARPA
Defense Advanced Research Projects Agency

NTONC
National Transparent Optical Network Consortium

NORTEL NETWORKS
BUSINESS WITHOUT BOUNDARIES

iCAIR

NSF

Topics

- Limitations of Packet Switched IP Networks
- Why DWDM-RAM?
- DWDM-RAM Architecture
- An Application Scenario
- Current DWDM-RAM Implementation

Limitations of Packet Switched Networks

What happens when a TeraByte file is sent over the Internet?

- If the network bandwidth is shared with millions of other users, the file transfer task will never be done (World Wide Wait syndrome)
- Inter-ISP SLAs are “as scarce as dragons”
- DoS, route flaps phenomena strike without notice

Fundamental Problems

1) Limited control and isolation of Network Bandwidth

2) Packet switching is not appropriate for data intensive applications => substantial overhead, delays, CapEx, OpEx

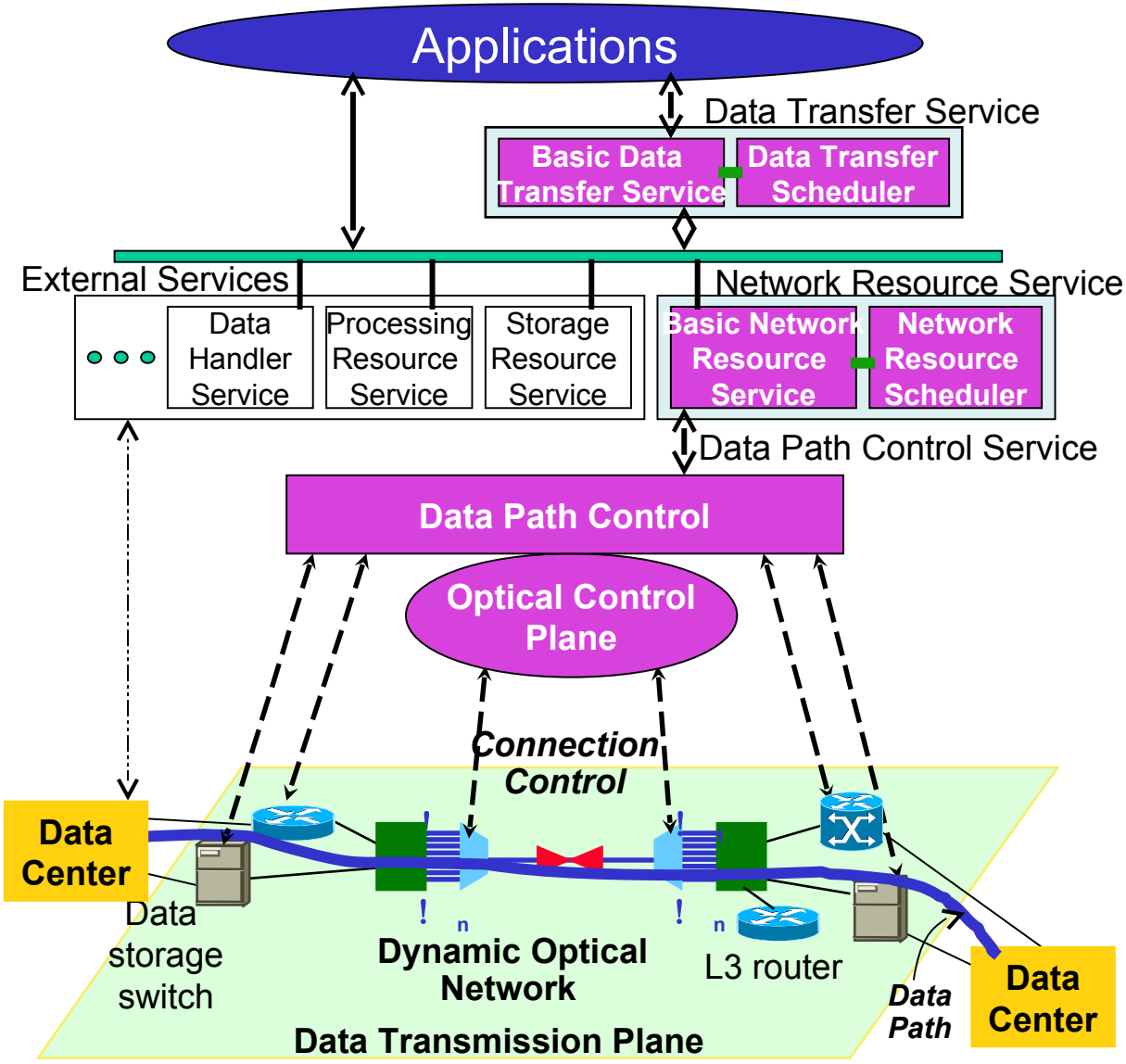
Why DWDM-RAM ?

- The new architecture is proposed for data intensive enabled by next generation dynamic optical networks
  - Offers a Lambda scheduling service over Lambda Grids
  - Supports both on-demand and scheduled data retrieval
  - Supports bulk data-transfer facilities using lambda-switched networks
  - Provides a generalized framework for high performance applications over next generation networks, not necessary optical end-to-end
  - Supports out-of-band tools for adaptive placement of data replicas

DWDM-RAM Architecture

- An OGSA compliant Grid architecture
- The middleware architecture modularizes components into services with well-defined interfaces
- The middleware architecture separates services into 3 principal service layers
  - Data Transfer Service Layer
  - Network Resource Service Layer
  - Data Path Control Service Layer over a Dynamic Optical Network

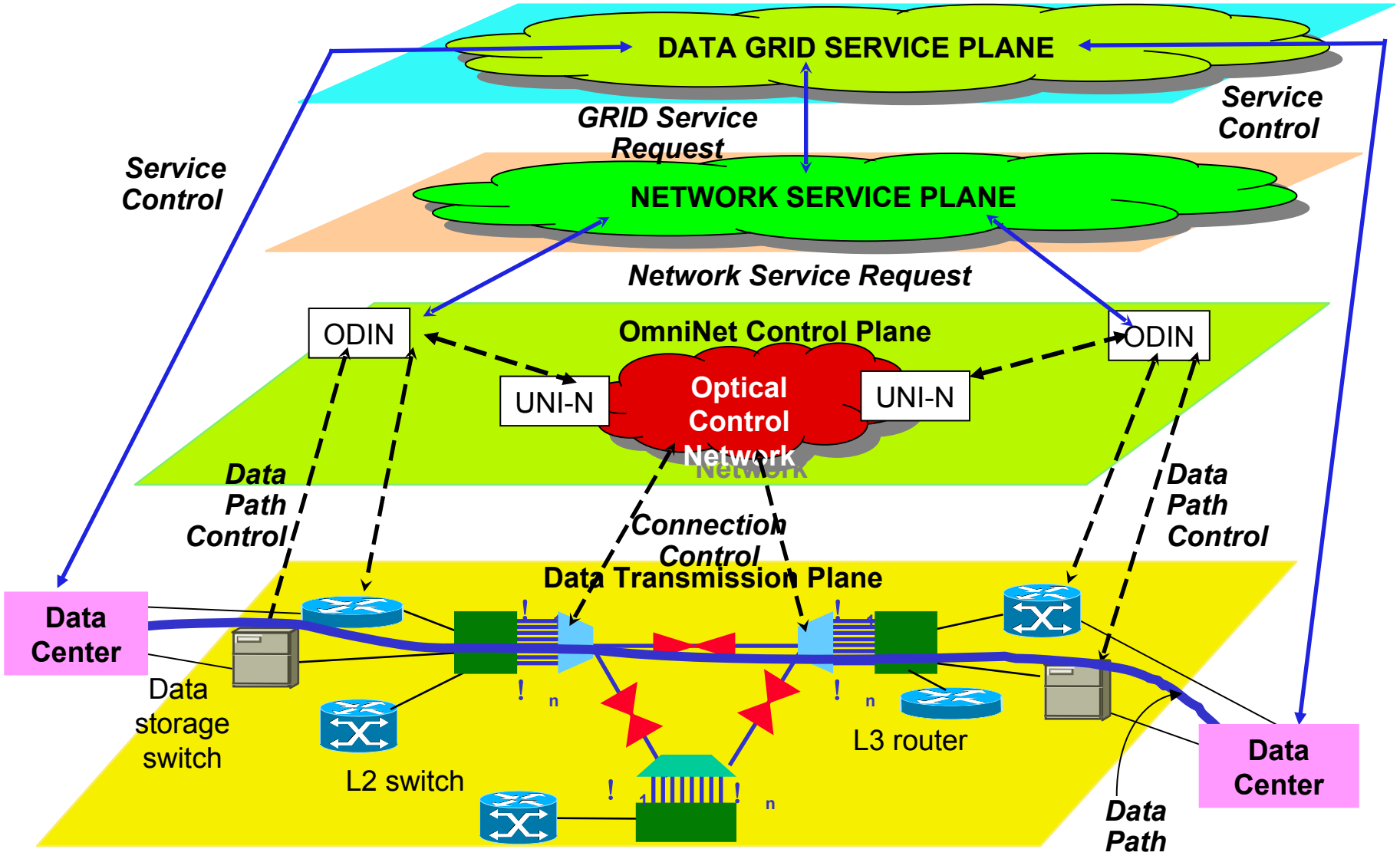
DWDM-RAM ARCHITECTURE



DWDM-RAM Service Architecture

- The Data Transfer Service (DTS):
  - Presents an interface between an application and a system – receives high-level requests, to transfer named blocks of data
  - Provides Data Transfer Scheduler Service: various models for scheduling, priorities, and event synchronization
- The Network Resource Service (NRS)
  - Provides an abstraction of "communication channels" as a network service
  - Provides an explicit representation of network resources scheduling model
  - Enables capabilities for dynamic on-demand provisioning and advance scheduling
  - Maintains schedules and provisions resources in accordance with the schedule
- Data Path Control Service Layer
  - Presents an interface between the network resource service and the network resources of the underlying network
  - Establishes, controls, and deallocates complete paths across both optical and electronic domains

DWDM-RAM Service Control Architecture



An Application Scenario: Fixed Bandwidth List Scheduling

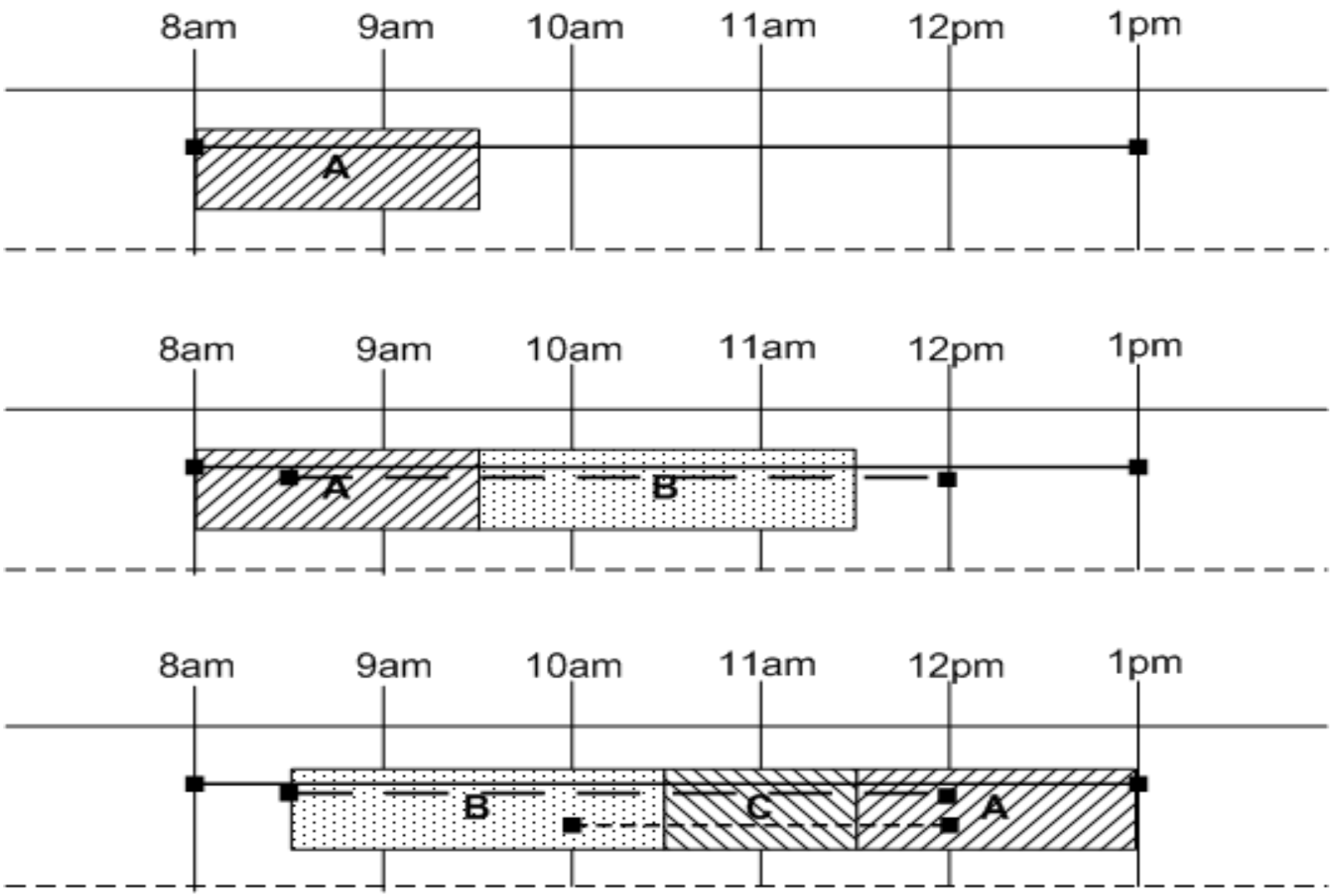
- A scheduling request is sent from the application to the NRS with the following five variables: Source host, Destination host, Duration of connection, Start time of request window, Finish time of request window
- The start and finish times of the request window are the upper and lower limits of when the connection can happen.
- The scheduler must then reserve a continuous hour slot somewhere within that time range. No bandwidth, or capacity, is referred to and the circuit designated to the connection is static.
- The message returned by the NRS is a “ticket” which informs of the success or failure of the request

Fixed Bandwidth List Scheduling

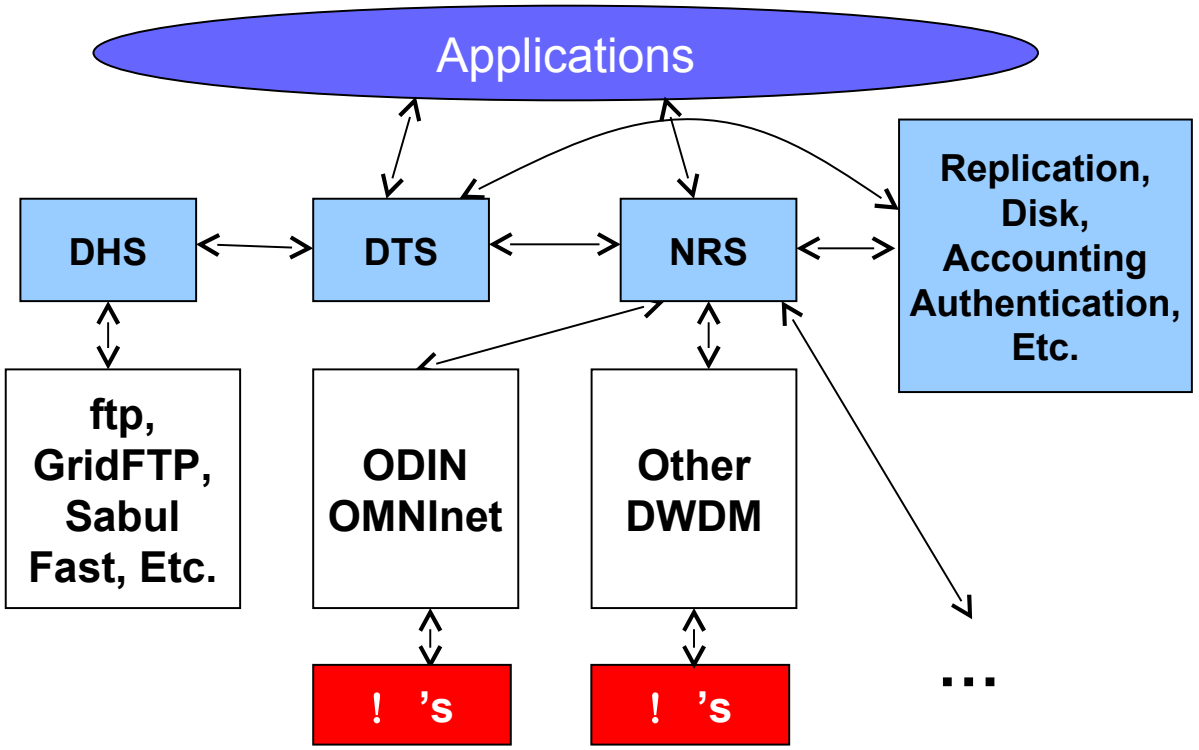
Job	**Job Run-time**	**Window**
A	1.5 hours	8am – 1pm
B	2 hours	8:30am – 12pm
C	1 hour	10am – 12pm

This scenario shows three jobs being scheduled sequentially, A, B and C. Job A is initially scheduled to start at the beginning of its under-constrained window. Job B can start after A and still satisfy its limits. Job C is more constrained with its runtime window but is a smaller job. The scheduler adapts to this conflict by intelligently rescheduling each job so all constraints are met.

Fixed Bandwidth List Scheduling



DWDM-RAM Implementation



Dynamic Optical Network

- Gives adequate and uncontested bandwidth to an application's burst
- Employs circuit-switching of large flows of data to avoid overheads in breaking flows into small packets and delays routing
- Is capable of automatic wavelength switching
- Is capable of automatic end-to-end path provisioning
- Provides a set of protocols for managing dynamically provisioned wavelengths

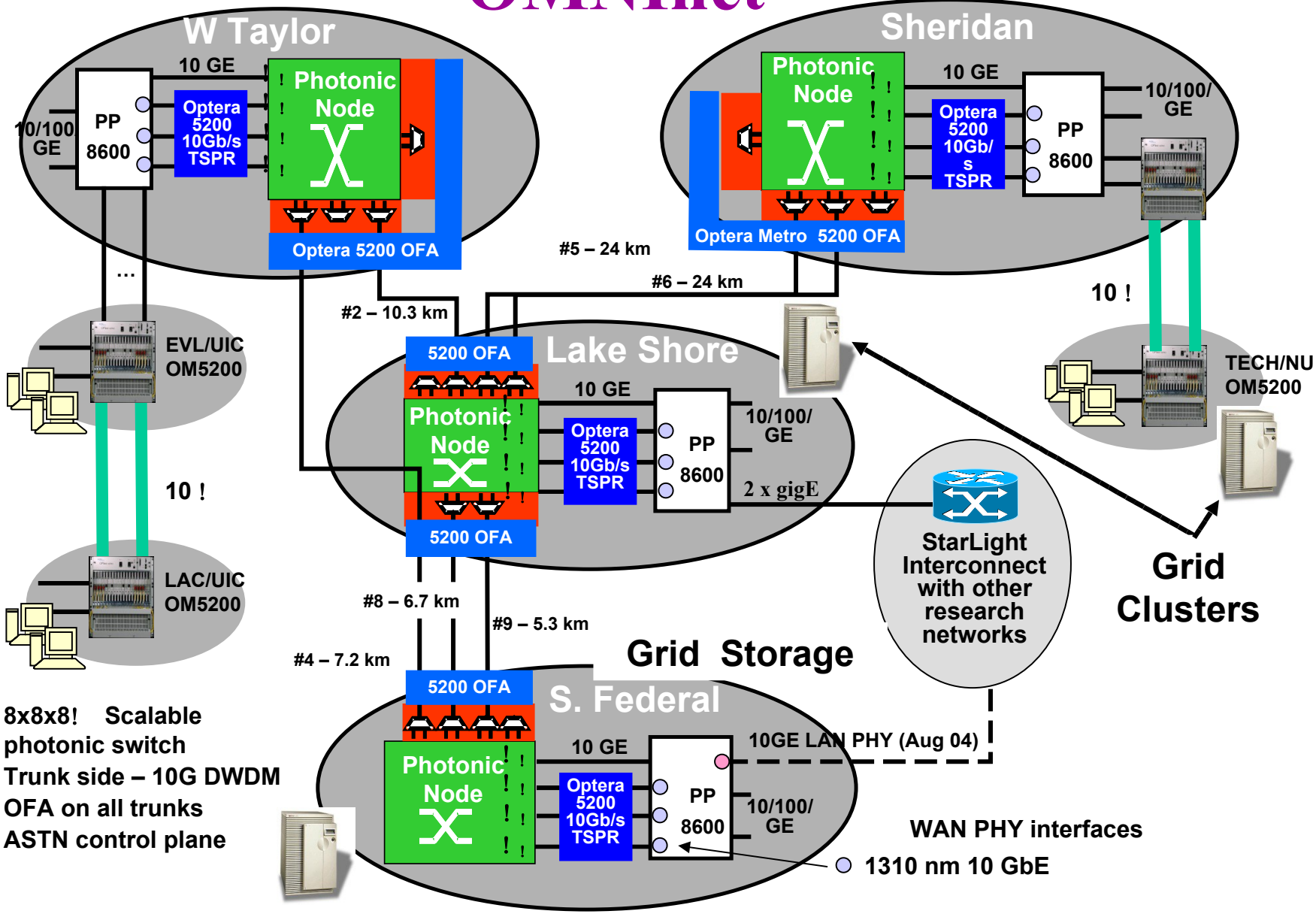
OMNInet Testbed

- Four-node multi-site optical metro testbed network in Chicago -- the first 10GigE service trial when installed in 2001
- Nodes are interconnected as a partial mesh with lightpaths provisioned with DWDM on dedicated fiber.
- Each node includes a MEMs-based WDM photonic switch, Optical Fiber Amplifier (OFA), optical transponders, and high-performance Ethernet switch.
- The switches are configured with four ports capable of supporting 10GigE.
- Application cluster and compute node access is provided by Passport 8600 L2/L3 switches, which are provisioned with 10/100/1000 Ethernet user ports, and a 10GigE LAN port.
- Partners: SBC, Nortel Networks, iCAIR/Northwestern University

Optical Dynamic Intelligent Network Services (ODIN)

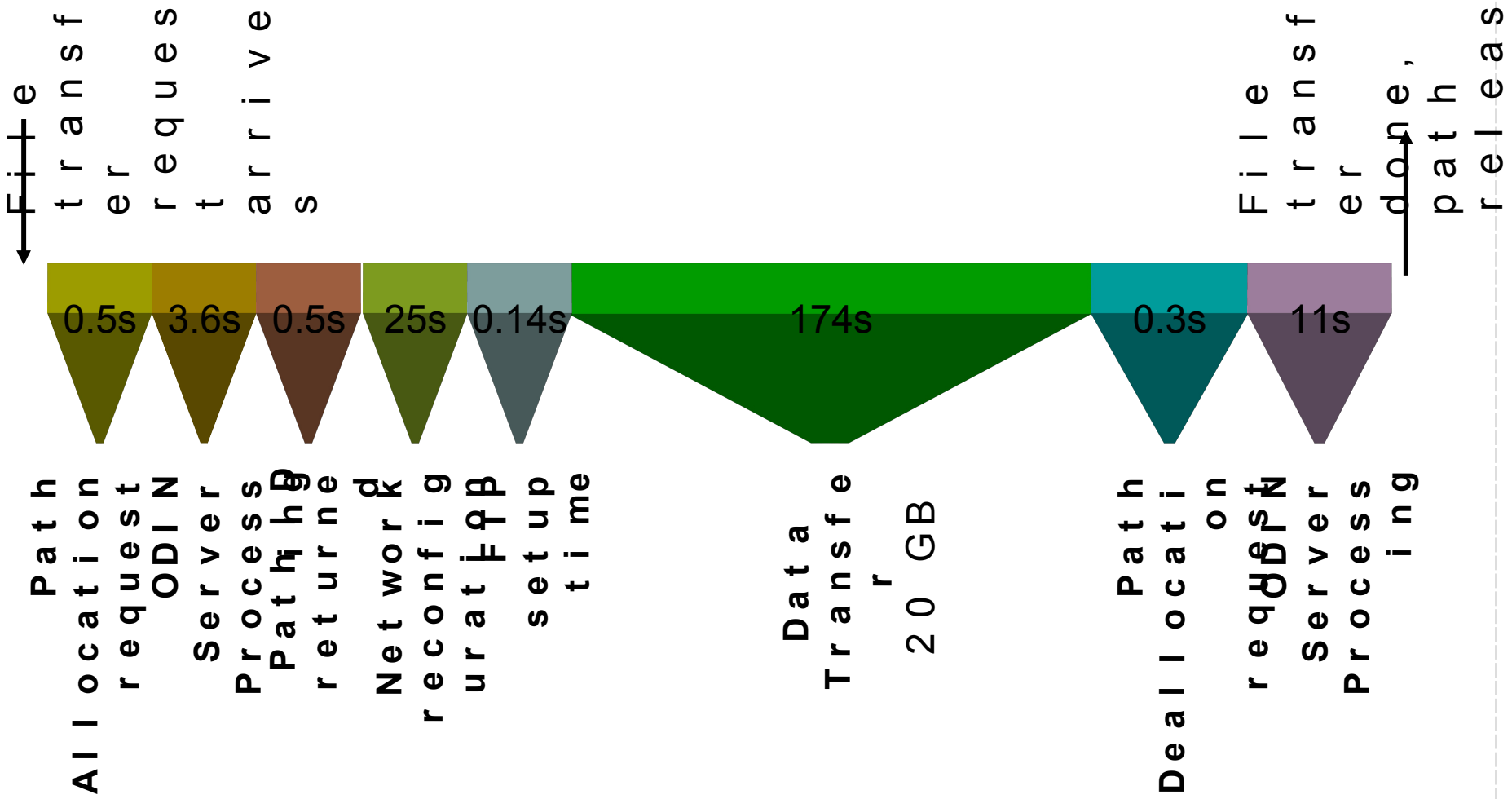
- Software suite that controls the OMNInet through lower-level API calls
- Designed for high-performance, long-term flow with flexible and fine grained control
- Stateless server, which includes an API to provide path provisioning and monitoring to the higher layers

OMNInet

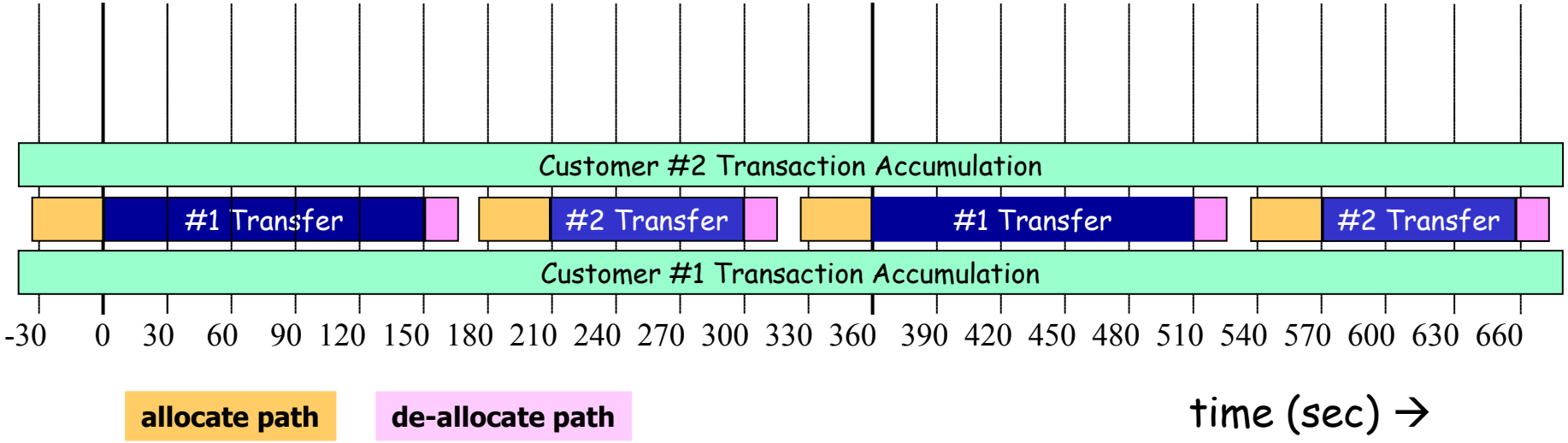


- 8x8x8! Scalable photonic switch
- Trunk side – 10G DWDM
- OFA on all trunks
- ASTN control plane

Initial Performance measure: End-to-End Transfer Time



Transaction Demonstration Time Line
6 minute cycle time



20GB File Transfer

Conclusion

- The DWDM-RAM architecture yields Data Intensive Services that best exploit Dynamic Optical Networks
- Network resources become actively managed, scheduled services
- This approach maximizes the satisfaction of high-capacity users while yielding good overall utilization of resources
- The service-centric approach is a foundation for new types of services

Some key folks checking us out at our CO2+Grid booth, GlobusWORLD '04, Jan '04

Ian Foster and Carl Kesselman, co-inventors of the Grid (2nd,5th from the left)

Larry Smarr of OptIPuter fame (6th and last from the left)

Franco, Tal, and Inder (1th, 3rd, and 4th from the left)

Data@LIGHTspeed

Defense Advanced Research Projects Agency

National Transparent Optical Network Consortium

BUSINESS WITHOUT BOUNDARIES

Back up slides

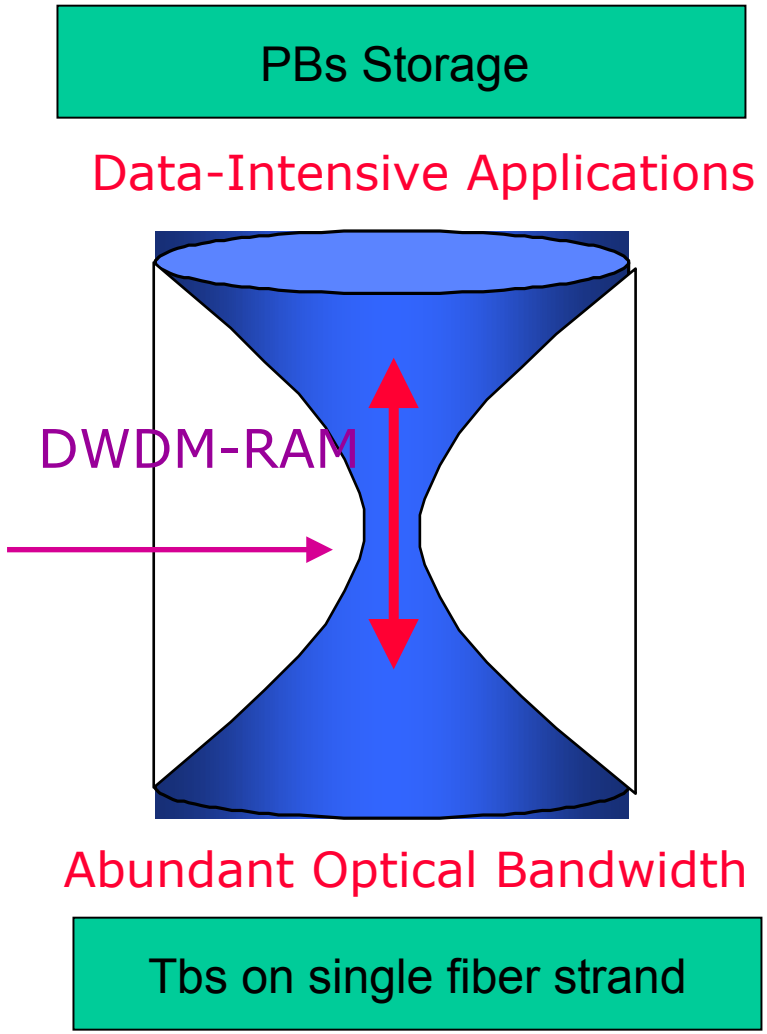
Optical Abundant Bandwidth Meets Grid

The Data Intensive App Challenge:

Emerging data intensive applications in the field of HEP, astro-physics, astronomy, bioinformatics, computational chemistry, etc., require extremely high performance and long term data flows, scalability for huge data volume, global reach, adjustability to unpredictable traffic behavior, and integration with multiple Grid resources.

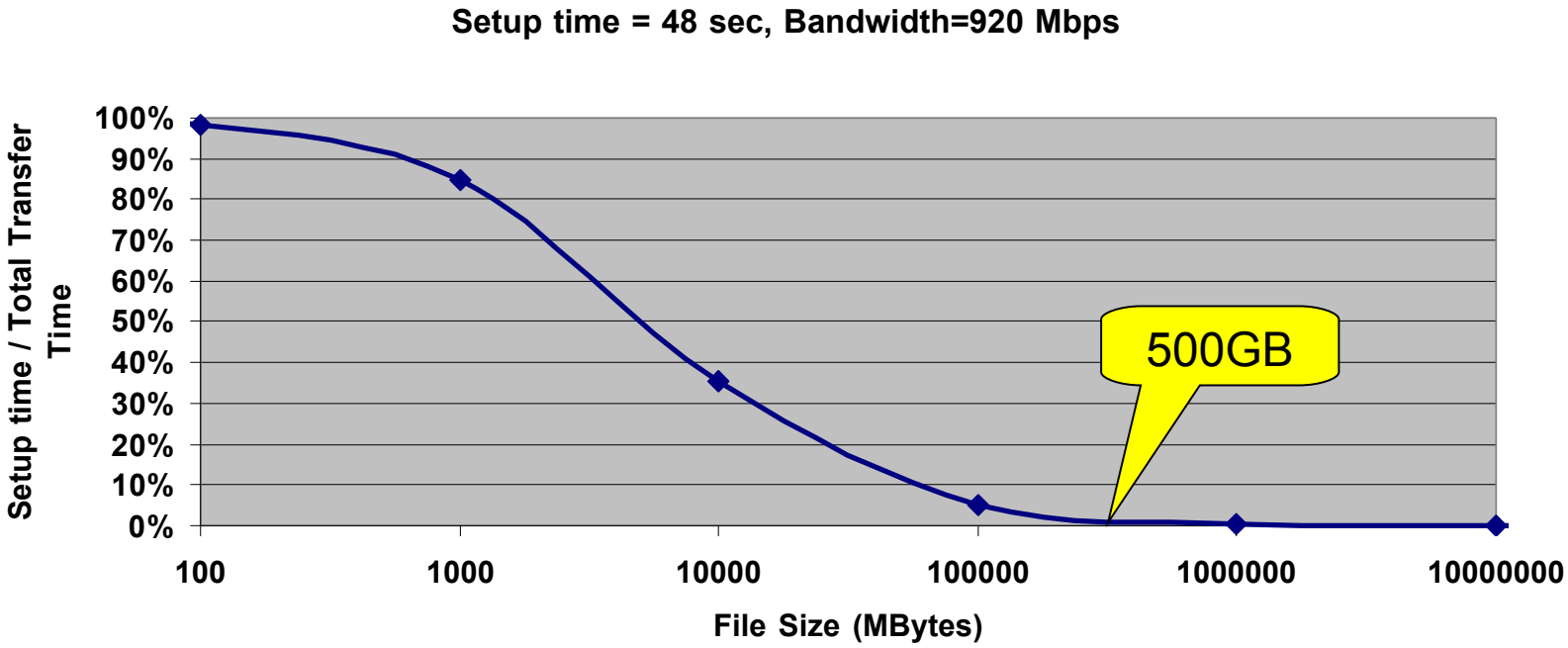
Response: DWDM-RAM

An architecture for data intensive Grids enabled by next generation dynamic optical networks, incorporating new methods for lightpath provisioning. **DWDM-RAM** is designed to meet the networking challenges of extremely large scale Grid applications. Traditional network infrastructure cannot meet these demands, especially, requirements for intensive data flows



Overheads - Amortization

When dealing with data-intensive applications, overhead is insignificant!



Grids urged us to think End-to-End Solutions

Look past boxes, feeds, and speeds

Apps such as Grids call for a complex mix of:

Bit-blasting

Finesse (*granularity of control)*

\+ Virtualization (*access to diverse knobs*)

\+ Resource bundling (*network AND …*)

\+ Multi-Domain Security (*AAA to start*)

\+ Freedom from GUIs, human intervention

SOFTWARE

Our recipe is a software-rich symbiosis of Packet and Optical products

NRS Interface and Functionality

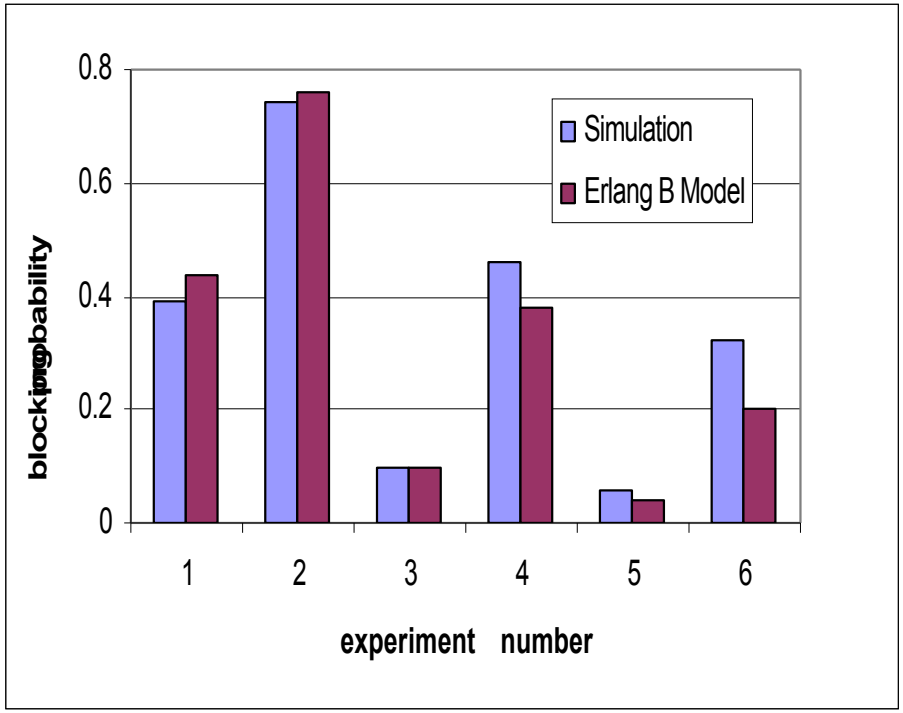
```
// Bind to an NRS service:
NRS = lookupNRS(address);
//Request cost function evaluation
request = {pathEndpointOneAddress,
           pathEndpointTwoAddress,
           duration,
           startAfterDate,
           endBeforeDate};
ticket = NRS.requestReservation(request);
// Inspect the ticket to determine success, and to find
the currently scheduled time:
ticket.display();
// The ticket may now be persisted and used
from another location
NRS.updateTicket(ticket);
// Inspect the ticket to see if the reservation's scheduled time has changed, or
verify that the job completed, with any relevant status information:
ticket.display();
```

Application Level Measurements

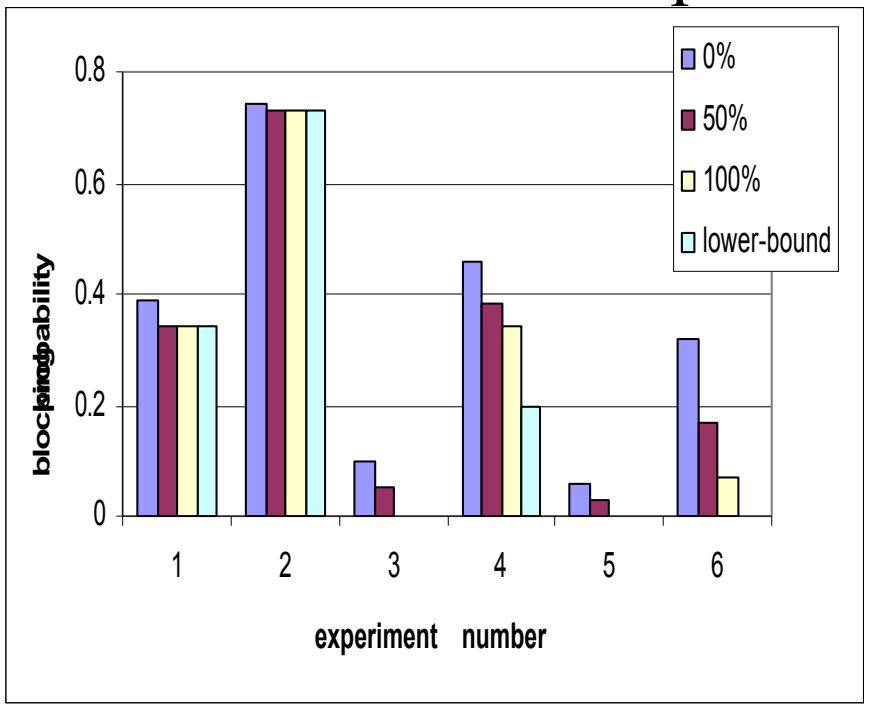
File size:	20 GB
Path allocation:	29.7 secs
Data transfer setup time:	0.141 secs
FTP transfer time:	174 secs
Maximum transfer rate:	935 Mbits/sec
Path tear down time:	11.3 secs
Effective transfer rate:	762 Mbits/sec

Network Scheduling – Simulation Study

Blocking Probability



Blocking probability Under-constrained requests



The Network Resource Service (NRS)

- Provides an OGSI-based interface to network resources
- Request parameters
  - Network addresses of the hosts to be connected
  - Window of time for the allocation
  - Duration of the allocation
  - Minimum and maximum acceptable bandwidth (future)

The Network Resource Service

- Provides the network resource
  - On demand
  - By advance reservation
- Network is requested within a window
  - Constrained
  - Under-constrained