StorNet Project
Status Review

BNL

August 30-31, 2010

Outline

- Project overview
- Design and integration aspects
- Theoretical aspects
- Year 1 status and Accomplisments
- Year 2 plans

BERKELEY LAB

BROOKHAVEN NATIONAL LABORATORY

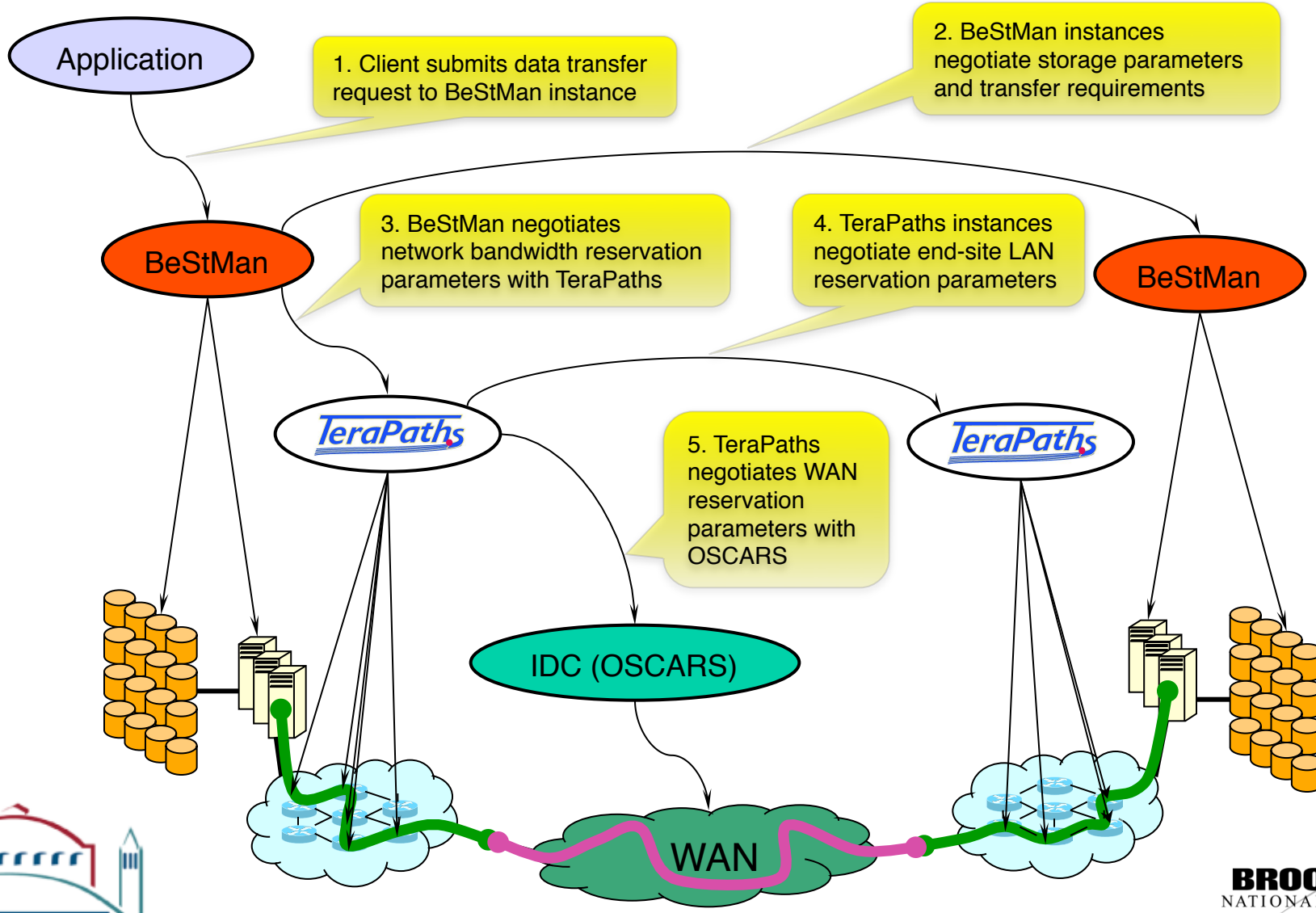
StorNet Project Overview

- Project Goals:
  - Design and develop an integrated end-to-end resource provisioning system for high performance data transfers
  - Improve resource utilization by co-scheduling network and storage resources and ensure data transfer efficiency
  - Support end-to-end data transfers with a negotiated transfer completion timeline.
  - Scheduling Network and Storage as a 1st Class Resource through Virtualization
  - Provide a holistic approach for DOE data-intensive applications to share data
  - Provide data management capabilities commensurate with exascale computing

BROOKHAVEN
NATIONAL LABORATORY

StorNet: System Design and Implementation

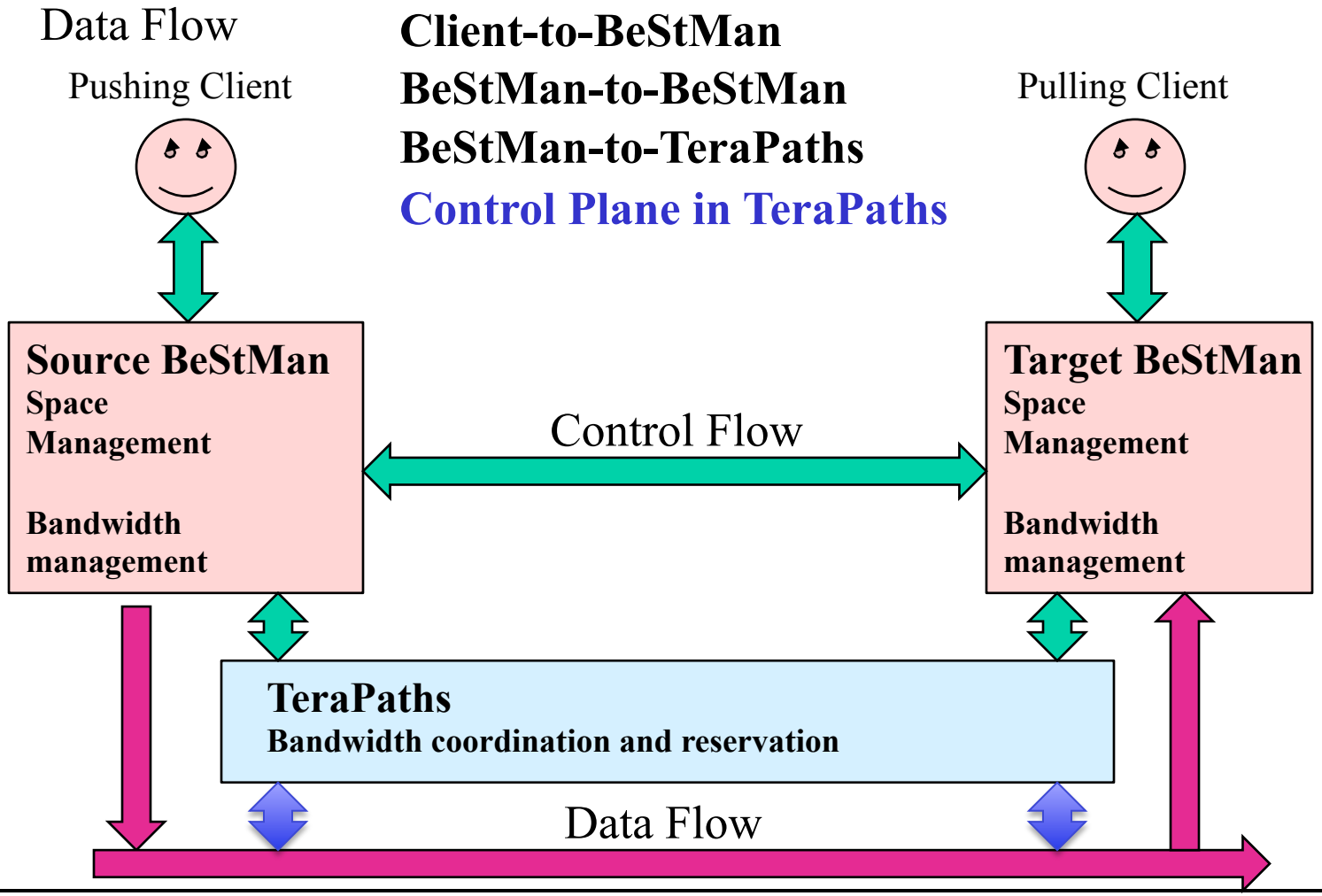
Integration of TeraPaths with BeStMan (SRM)



BERKELEY LAB

BROOKHAVEN NATIONAL LABORATORY

New APIs to be defined and functionality and Communication Flow developed



Enhancements Needed for StorNet

- BeStMan needs enhancements to:
  - Keep track of bandwidth commitments for multiple request
    - Both Storage and Network bandwidths
    - Backend database support
  - Coordinate between source and target BeStMan's for storage space and bandwidth
  - Provide advanced reservation for future time window commitments
  - Communication and coordination with underlying TeraPaths
- TeraPaths needs enhancements to:
  - Receive network bandwidth requests from BeStMan with inputs (volume, max-bandwidth, max-completion-time)
  - Negotiate with OSCARS for "best" time window
    - "best" can be earliest completion time, or shortest transfer time
  - If success, return to BeStMan and commit reservation if BeStMan desires.
  - If failure, find closest solution to suggest to BeStMan

BeStMan-TeraPaths API

- Main functions:
  - reserveRequest()
    - Input: flow specs (source/destination IPs and ports), bandwidth, start time, end time, transfer volume
    - Output: request token, reservation ids
  - commitRequest()
    - Commits the network reservation.
  - cancelRequest()
- Auxiliary functions
  - statusRequest()
  - extendTimeoutRequest()
    - Extends timeout if additional time is needed before committing
  - modifyRequest()
    - Modifies request parameters – primarily needed when flow specs are not known at time of reserve request

BROOKHAVEN
NATIONAL LABORATORY

β

TeraPaths Database Design

- Requests
  - From client, such as BeStMan
  - RequestID, protocol, timeout, srcIP, destIP, etc.
- E2EReservations
  - Consisting of two LAN reservations and WAN reservation
  - startTime, duration, bandwidth, etc.
- DomainReservations
  - LAN reservation at local domains or WAN reservation (OSCARS) at transit domain
  - startTime, duration, bandwidth, type, status, etc.

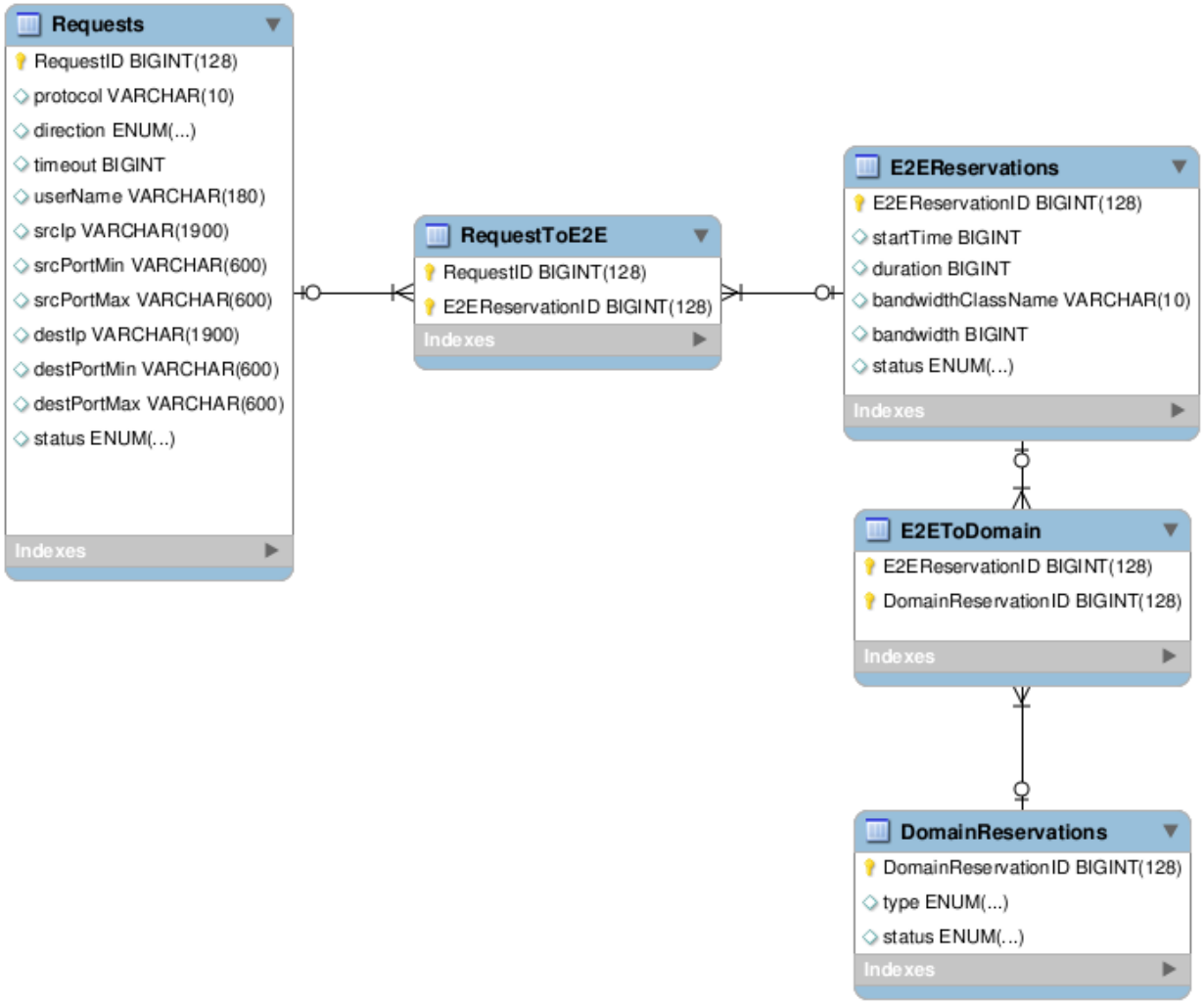
BERKELEY LAB

BROOKHAVEN NATIONAL LABORATORY

TeraPaths Database Design (ii)

- Many to many relation between Requests and E2EReservations
  - Simplest case, one request is satisfied by one end-to-end reservation
  - To improve utilization, one request could be satisfied by several transfers in separate time frames
  - One end-to-end reservation can serve multiple requests (consolidation)
- Many to many relation between E2EReservations and DomainReservations
  - Normally, one E2EReservation consists of two LAN reservations at end sites and one WAN reservation
  - Special case, one transit domain reservation (OSCARS circuit) can serve multiple reservation (semi-static circuits)
- Use junction tables to break many-to-many to one-to-many

Core TeraPaths DB schema

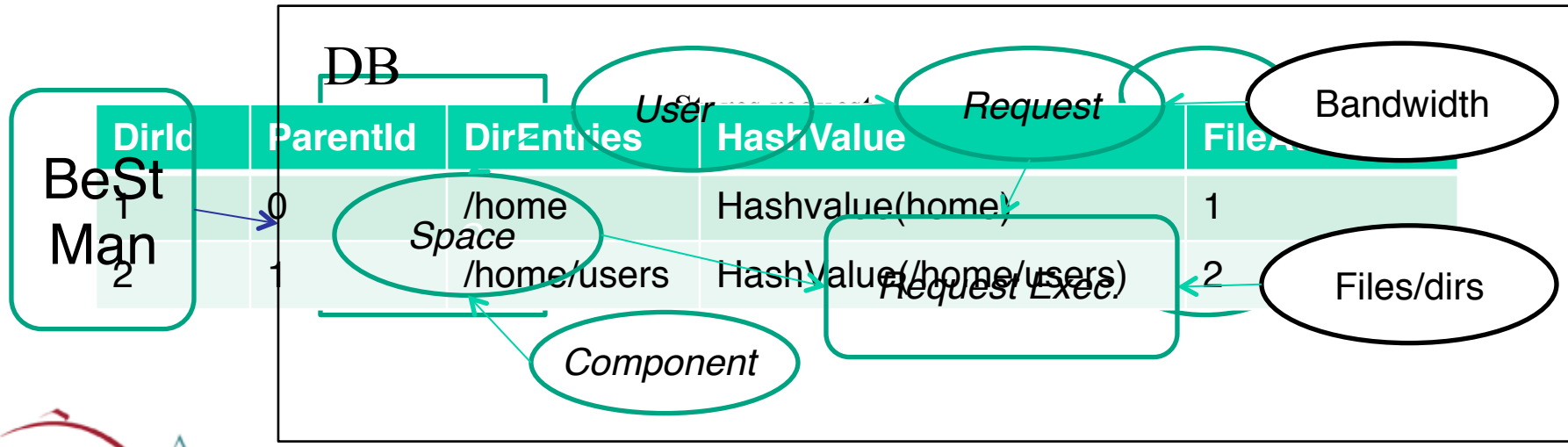


BeStMan Services in StorNet

- SRM storage services:
  - Processing storage service request, and subsequent coordinating network planes.
- Network reservation services:
  - Reservation end-to-end circuits connecting two storage places through TeraPaths.
- Request state/status report:
  - Monitor SRM data transfer progress and performance.
  - Monitor end-to-end circuit state and performance.

BeStMan Backend DB Support

- DB access from BeStMan
  - Store, retrieve and update the requests
  - Asynchronous requests to the DB
  - Multiple BeStMan instances can access the DB for load balancing on user request handling.
  - Performance and efficiency in DB access
    - Use of hash values



BERKELEY LAB

BROOKHAVEN NATIONAL LABORATORY

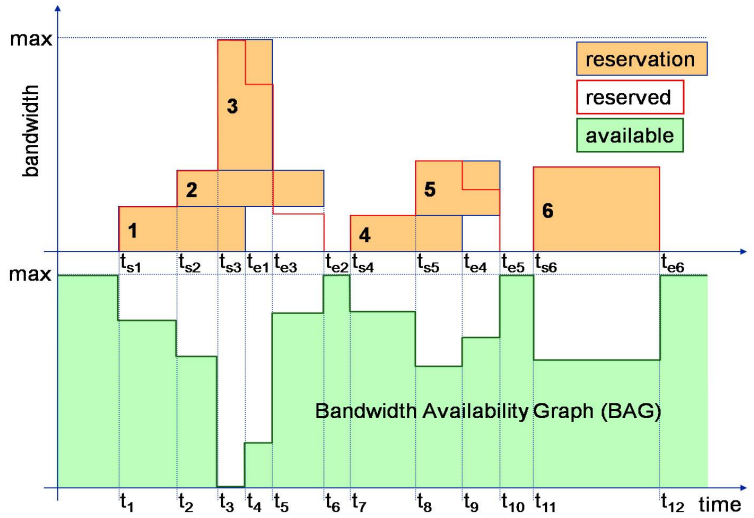
Reservation Negotiation: Overview

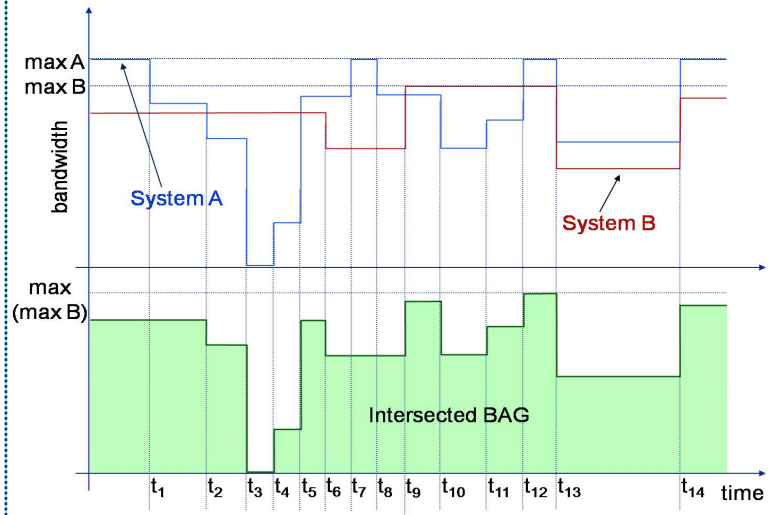
- Between BeStMan and TeraPaths
  - BeStMan will send a (storage) Bandwidth Availability Graph (BAG) to TeraPaths along with the request
- Between TeraPaths end sites
  - The master TeraPaths resource manager will gather BAGs from two LANs at local end sites
  - Intersect all BAGs (from both BeStMan and TeraPaths) to form a “local” BAG
- Between TeraPaths and OSCARS
  - Obtain a ordered list of *best* reservations from the “local” BAG and send them to OSCARS
- Once a reservation can be made in OSCARS
  - OSCARS->TeraPaths->BeStMan

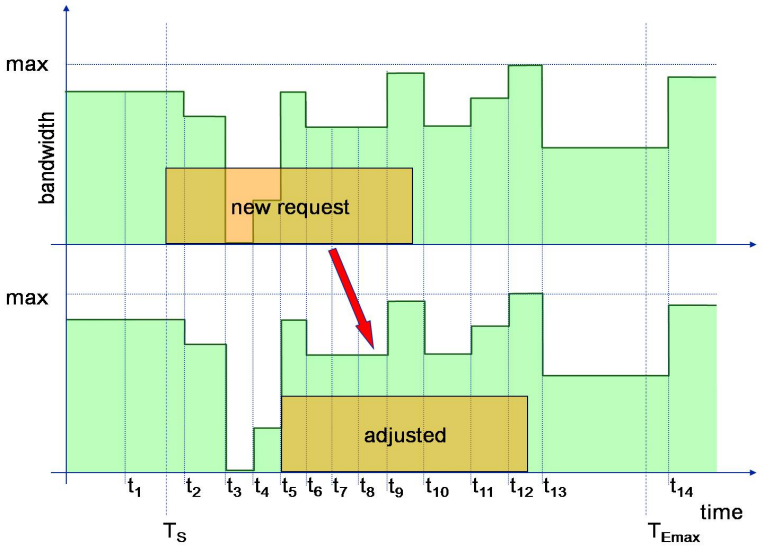
Reservation Negotiation: Algorithms

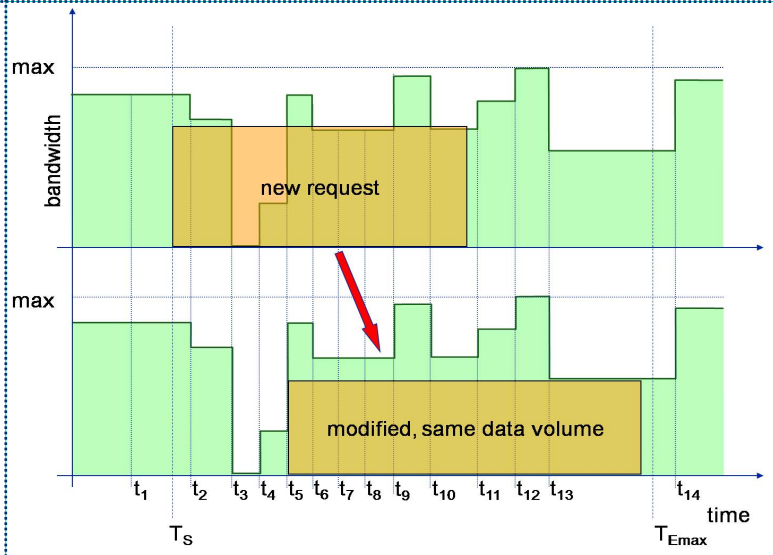
- Individual BAG can be obtained in a linear time
  - Get all reservations in [min start time, max finish time]
  - Group them and get bandwidth usage graph (BUG)
  - Subtract BUG from the capacity and get BAG
- Intersecting BAGs can be done in a linear time
  - Similar to above

Reservation Negotiation: Algorithms (ii)







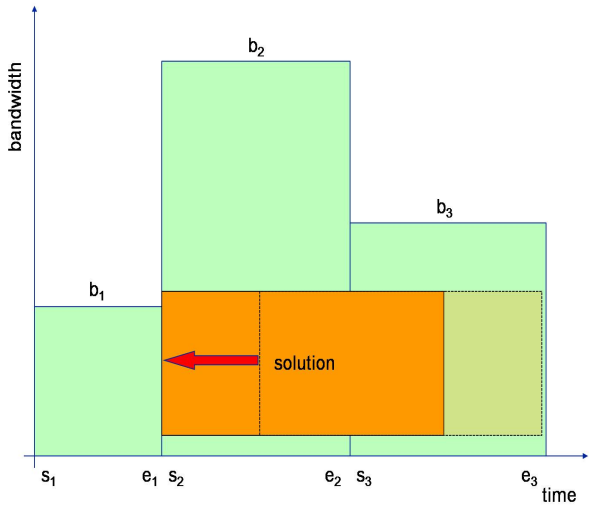


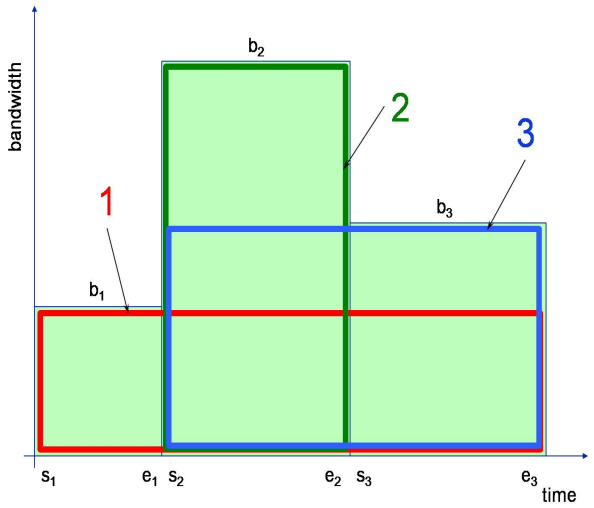
BERKELEY LAB

BROOKHAVEN NATIONAL LABORATORY

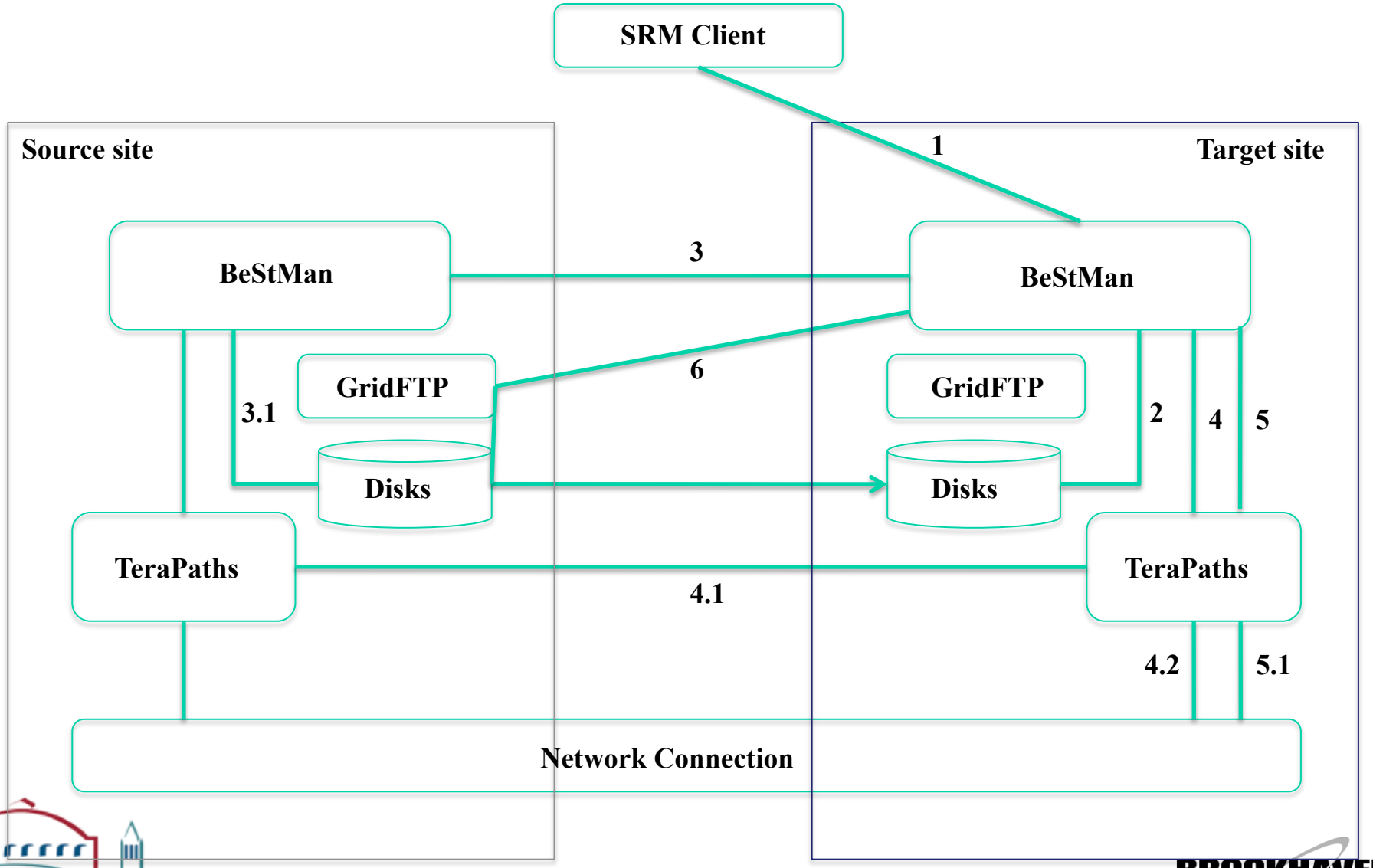
Reservation Negotiation: Algorithms (iii)

- Finding optimal solutions can be done in a linear time
  - Objectives : shortest duration or earliest finish time
  - Similar to problem of “The Largest Rectangle under a Histogram”
  - For each availability segment, find the largest rectangle containing it
  - Then select best solutions from those largest rectangles





Workflow in StorNet

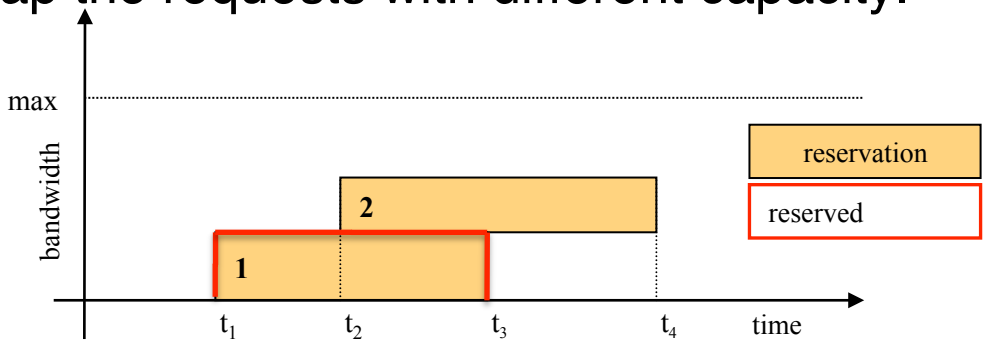


Year 1 Status

- Year 1:
  - Analysis of storage/network co-scheduling requirements
  - BeStMan/TeraPaths integration: design of StorNet API
  - Design of necessary enhancements to BeStMan and TeraPaths
  - Implementation of StorNet API
  - Deployment and testing of basic functionality on the BNL/UMich TeraPaths testbed

Initial test cases

- Simple request (Single file in a request)
  - A request that contain one file that needs to be copied from the source to the target. Storage needs to be allocated for the file, and bandwidth needs to be reserved between the sites.
  - This test was done successfully.
- Large request (Multiple files in a request)
  - This test case consists of many files in a request. The files need to be copied from the source to the target together.
  - This test is on going.
- Multiple requests
  - A multiple requests test case consists of mixture of small and large requests, so that both storage and network bandwidth reservations can overlap the requests with different capacity.



Year 2 Plan

- Implementation:
  - Additional support required for aux. calls
    - modifyRequest()
    - statusRequest()
    - extendTimeoutRequest()
  - Detection/resolution of flow spec conflicts
  - Reservation negotiation
  - Support for multiple-window requests
- Testbed:
  - Deploy at LBNL
- Testing:
  - Multiple transfers with overlapping windows