

Research Directions in Network Services

Han Gao

University of Chicago

AG Technical



Outline

- Introduction
- Matchmaker
- Multicast Monitoring:
 - Multicast Clustering
 - Fairness of Multicast
- Access Grid Integration
- Several Interesting Network Services
- Summary



Introduction

- Access Grid Assumption:
 - Higher network bandwidth;
 - Higher computational capability;
- Constraints of Access Grid:
 - Location;
 - Resource (bandwidth, computing power, hardware);



Introduction

- Goals of Network Services:
 - Free location constraints;
 - Relaxing resource constraints:
 - Integrated with various network bandwidth;
 - Compatible with different computing capability;
 - Portable to multi-platform

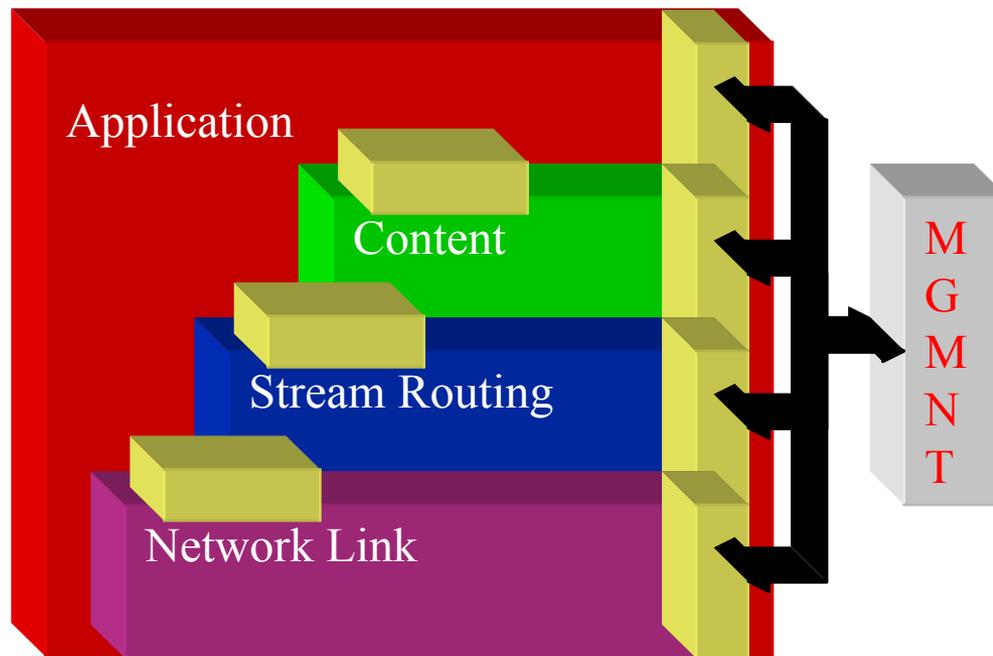


Introduction

- Network Services Architecture:
 - Soft-state
 - Decentralization
 - Self-adaptable
 - Transparent
 - Stream-Capability
- Two basic management modules:
 - Network Services Matchmaker
 - Multicast Monitoring



Introduction (NSA)



Matchmaker

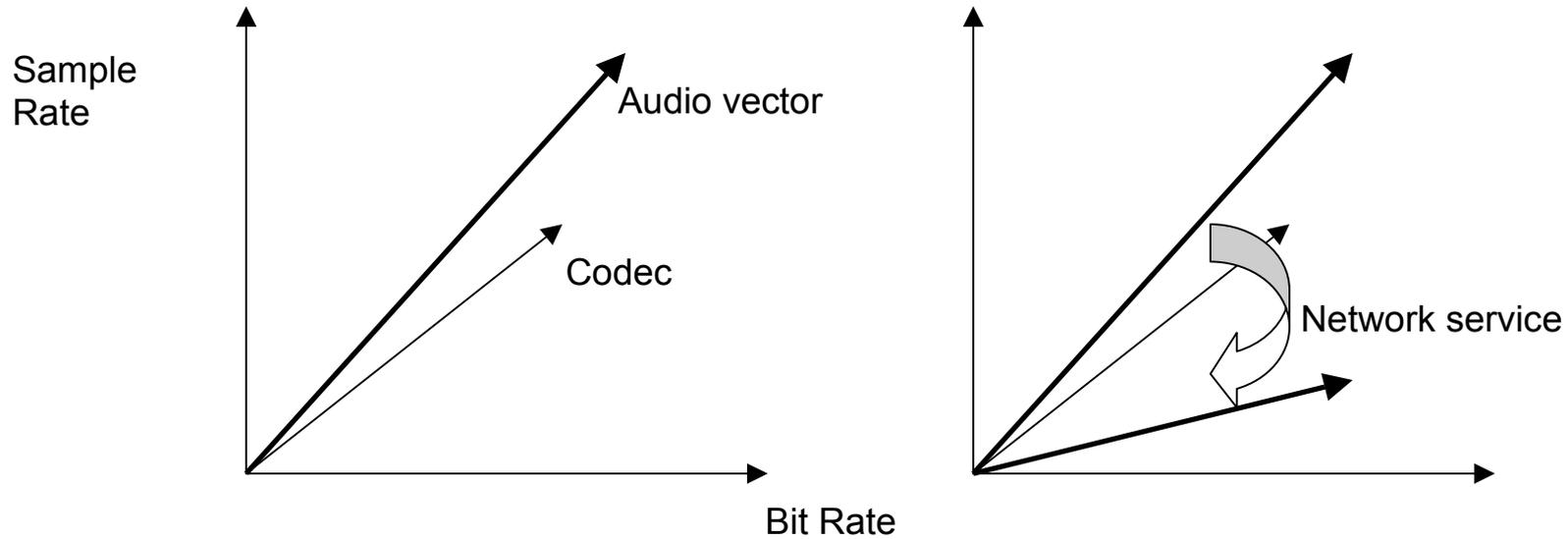
- Service Management
 - Describing
 - Publishing
 - Discovering
 - Registering
 - Selecting (matching)
- Two-party vs. Three party



Matchmaker –mathematical definitions

- Definition 1: A capability vector space over \mathbb{R}^n is a set of capability vectors for which the dimension of any capability vector is n .
- Definition 2: A capability vector is an element of a capability vector space. In the commonly encountered capability vector space \mathbb{R}^n , a capability vector is given by n coordinates and can be specified as $X=(x_1, x_2, \dots, x_n)$
- Definition 3: The operation of a stream transformation is the operation of a transformation matrix in capability space that transformed the stream capabilities. A transformation matrix, T , is a concise and useful way of uniquely representing and working with stream transformations:

Matchmaker

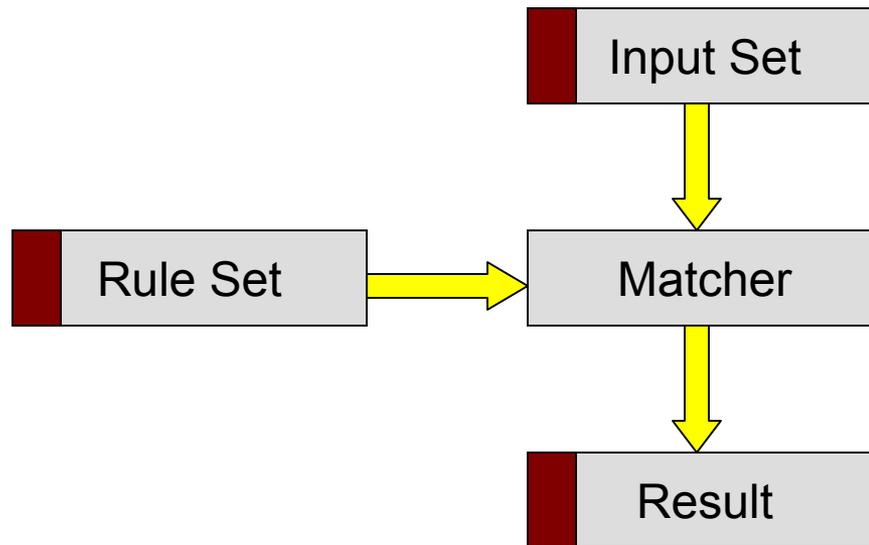


Matchmaker --General Method

Algorithm : Given transformation matrices, searching for a largest common capability vector or a set of common capability vectors and a set of transformation matrices that build the largest common vector or common vector set.

$$v^* = \arg \max \{v | v \in S^*\}$$
$$T^* = \{T | v^* = Tv\}$$

Matchmaker -- architecture



Matchmaker –user description

- XML-ClassAd-Schema (XCS) for Data Stream
 - ID: ID of group member who has a set of stream capabilities.
 - TYPE: Multimedia type of the stream.
 - DESCRIPTION: Explanation of terms used in the schema, including the description of stream and context. This entry is designed to be understood easily. It is not to be read, parsed, or compiled by the matchmaker.
 - PREFERENCE: List of preferred stream capabilities. The preference depends on the conditions and constraints of the user's facilities or the user's personal choice.
 - RESOLUTION: List of user-available stream capabilities. There is no special order for these. The items in the preference entry are a subset of those in the resolution entry.

Matchmaker —user description example

```
<c>
<a n="ID"><s>audio.cs.uchicago.edu</s></a>
<a n="type"><s>audio</s></a>
<a n="description"><s>audio stream capability ... </s></a>
<a n="preference"></a>
  <c>
    <a n="name"><s>L1 6-8K-Mono</s></a>
    <a n="codec"><s>Linear-16</s></a>
    <a n="sample_rate"><n>8000</n></a>
    <a n="bit_rate"><n>128.0</n></a>
    <a n="channel"><n>1</n></a>
  </c></a>
<a n="resolution"></a>
  <c>
    <a n="name"><s>L1 6-8K-Stereo</s></a>
    <a n="codec"><s>Linear-16</s></a>
    <a n="sample_rate"><n>8000</n></a>
    <a n="bit_rate"><n>256.0</n></a>
    <a n="channel"><n>2</n></a>
  </c>
  .....
```

Matchmaker --Network Service Description

- XML-ClassAd-Schema (XCS) for Network Service
 - ID: ID of the network service provider.
 - TYPE: Type of network service.
 - DESCRIPTION: Ontological description of network service, and explanation of terms used.
 - OUTPUT/INPUT: Name of the specified output or input stream.
 - OUTPUT/INPUT CODEC: Encoding or decoding algorithm for the output or input stream.
 - OUTPUT/INPUT SAMPLE_RATE: Sampling frequency of the output or input stream.
 - OUTPUT/INPUT BIT_RATE: Bandwidth cost of the output or input stream.
 - OUTPUT/INPUT CHANNEL: Sound effect of the output or input stream.
Usually, more channels produce a better sound effect.

Matchmaker —network service description example

<c>

```
<a n="ID"><s>networkservice.l.gawaine.cs.uchicago.edu</s></a>
```

```
<a n="type"><s>audioService</s></a>
```

```
<a n="description"><s></s></a>
```

```
<a n="output"><s>L16-8K-Mono</s></a>
```

```
<a n="outputcodec"><s>Linear-16</s></a>
```

```
<a n="outputsample_rate"><n>8000</n></a>
```

```
<a n="outputbit_rate"><n>128.0</n></a>
```

```
<a n="outputchannel"><n>1</n></a>
```

```
<a n="input"><s>L16-16K-Mono</s></a>
```

```
<a n="inputcodec"><s>Linear-16</s></a>
```

```
<a n="inputsample_rate"><n>16000</n></a>
```

```
<a n="inputbit_rate"><n>256.0</n></a>
```

```
<a n="inputchannel"><n>1</n></a>
```

</c>

Matchmaker --implementation

- Implementing one to many and many to many (set to set) matching algorithm
- gSoap implementation
 - Develop C++ code without thinking about restriction, type of definition and syntax matter of soap.
- One integrated function:

String matcher_match(list **inputs, int rules)*

Matchmaker --Results

- Audio capabilities for each users

User1: {(L16-8K-Mono, L16, 8K, 1)(preferred),
(L16-16K-Mono, L16, 16K, 1)};

User2: {(L16-8K-Mono, L16, 8K, 1)(preferred),
(L16-32K-Mono, L16, 32K, 1)};

User3: {(L16-8K-Mono, L16, 8K, 1)(preferred),
(L8-8K-Mono, L8, 8K, 1),
(L8-16K-Mono, L8, 16K, 1)};

User4: {(L16-8K-Mono, L16, 8K, 1)(preferred),
(G276-32-16K-Mono), (LPC-8K-Mono)}

Matchmake --Result

- Matching Result without network service:

The following preferred streams are available:

The stream L16-8K-Mono is available for all 4 users

Codec : Linear-16

Sample Rate : 8000Hz

Bit Rate : 128kbps

Channel : 1

All of the available streams as follows:

The stream L16-8K-Mono is available for all 4 users

Codec : Linear-16

Sample Rate : 8000Hz

Bit Rate : 128kbps

Channel : 1



Matchmaker --Results

- Introducing network services:

Network service 1:

$(L16-8K-Mono, L16, 8K, 1) \leftrightarrow (L16-16K-Mono, L16, 16K, 1)$

Network service 2:

$(L8-8K-Mono, L8, 8K, 1) \leftrightarrow (L16-16K-Mono, L16, 16K, 1)$

Matchmaker --Results

- Matchmaking with network services:

The following preferred streams are available:

The stream L16-8K-Mono is available for all 4 users

.....

All of the available streams as following:

*audio2.gawaine.cs.uchicago.edu:networkservice1.gawaine.cs.uchicago.edu:L16-16K-Mono <-->
L16-8K-Mono*

*audio3.gawaine.cs.uchicago.edu:networkservice1.gawaine.cs.uchicago.edu:L16-16K-Mono <-->
L16-8K-Mono*

*audio3.gawaine.cs.uchicago.edu:networkservice2.gawaine.cs.uchicago.edu:L16-16K-Mono <--> L8-
8K-Mono*

*audio4.gawaine.cs.uchicago.edu:networkservice1.gawaine.cs.uchicago.edu:L16-16K-Mono <-->
L16-8K-Mono*

The stream L16-16K-Mono is available for all 4 users

.....

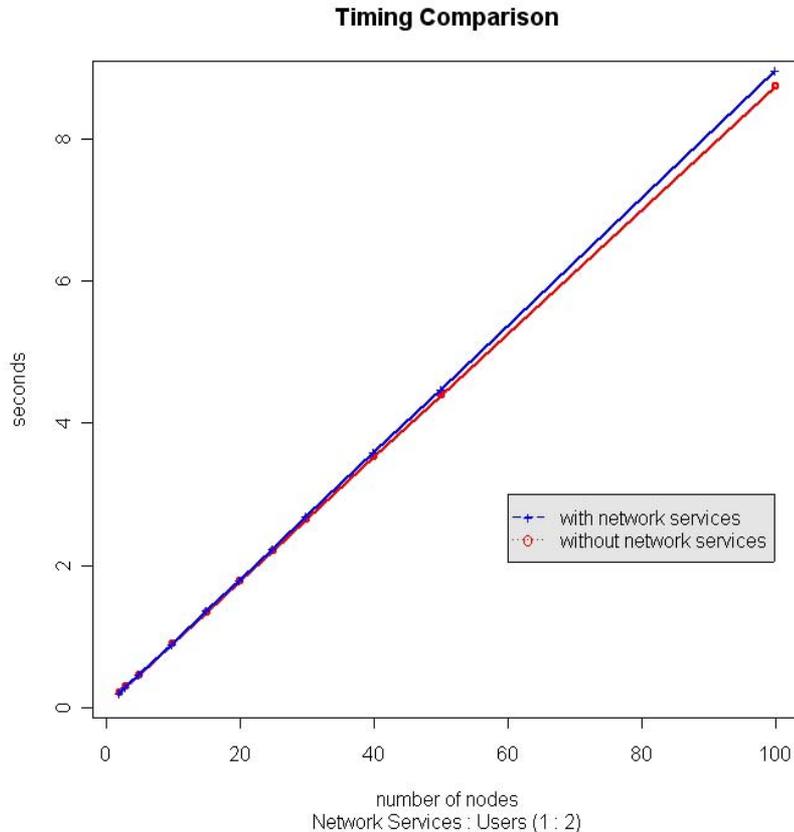
The stream L16-8K-Mono is available for all 4 users

.....



Matchmaker --performance

- Performance analysis:
 - Pentium 4 1.66 Hz, 512 MB memory, RedHat 7.3
 - Timing comparison between matching with network service and without network service. The number of nodes ranges from 2 to 100.
 - Computational complexity: $O(n)$



Multicast Monitoring

- Where we should put network services into existing network topologies;
- predicting, monitoring and analyzing the logical multicast topology;
- knowledge of the multicast topology could be used by multicast applications;

Multicast Clustering

- Data Collection:
 - Loss -- percentage of packet loss from one host to another
 - Delay -- one-way delay from one host to another
 - Jitter – variation of the one-way delay
 - Order -- percentage of packets which arrived out-of-order
 - Duplicate -- percentage of duplicated packets

Multicast Clustering

- Hierarchical: from source to final receivers, data streams are always transferred through several routers and network layers. Optimal multicast tree has multiple levels of hierarchy.
- Local: Due to the geographical factors, users in the same local area seem have very good connectivity and share some common characteristics.
- Dynamic: each distributed user participates or leaves one multicast session freely

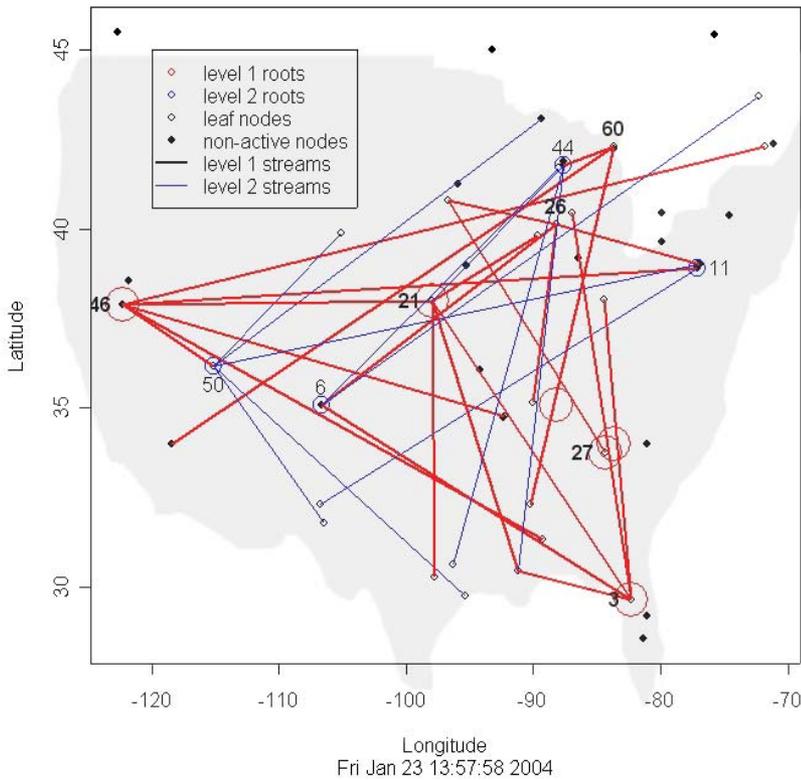
Multicast Clustering

- Hierarchical K-means Clustering:

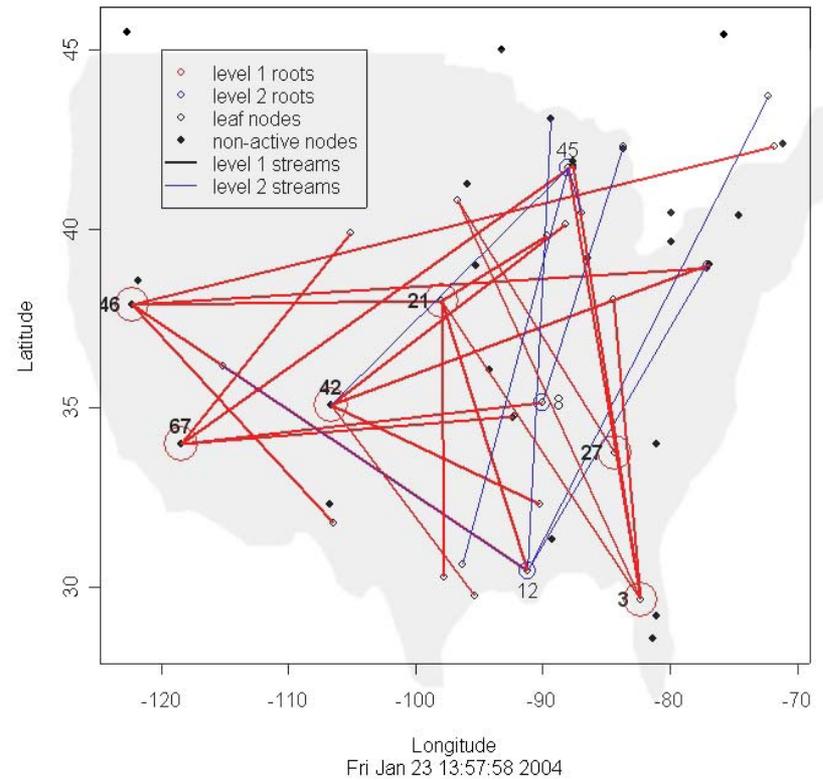
```
def hkc(nodes)
  clusterSize = n;
  seeds = initializeSeeds(nodes,n);
  clusters = k_means(nodes, seeds, n);
  for cluster in clusters
    if length(cluster) < n then
      hkc(cluster)
    else
      output(cluster)
  output(clusters)
```

Multicast Clustering

Logical Topology of Access Grid Nodes in North America



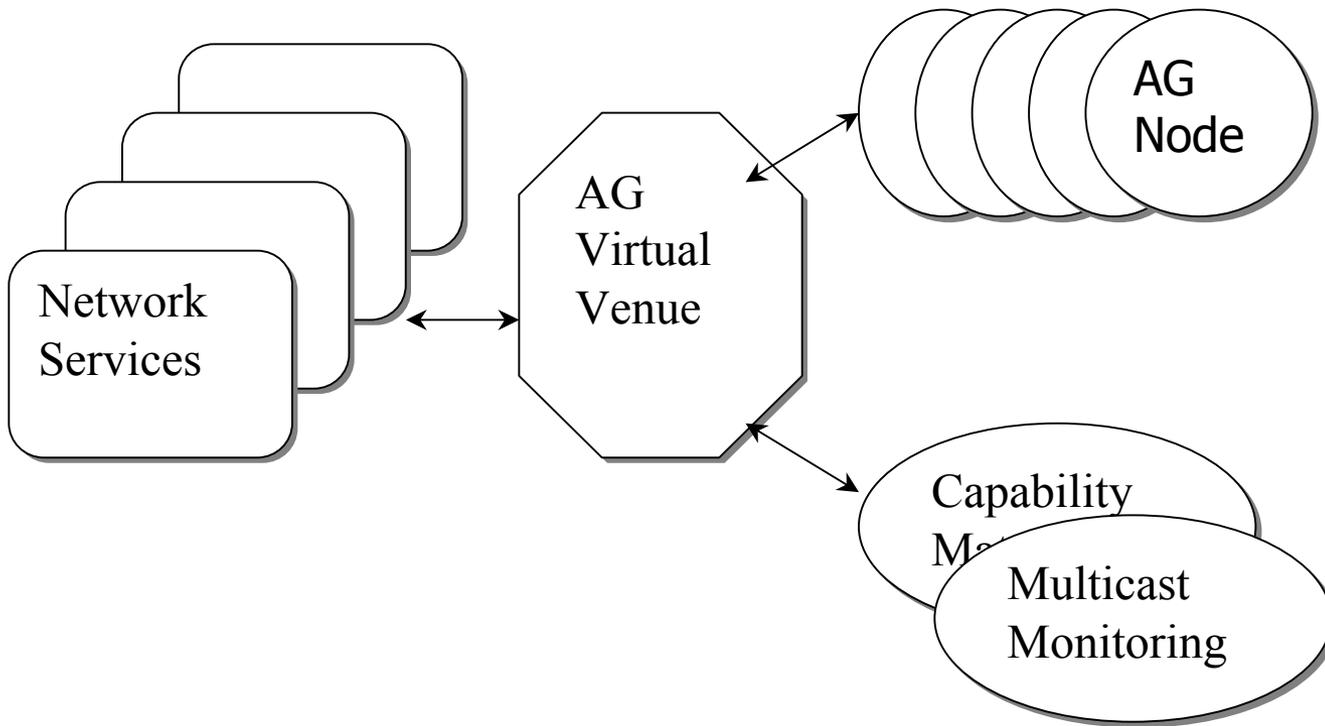
Logical Topology of Access Grid Nodes in North America



Fairness of Multicast

- a precise metric is necessary for further evaluation of performance, efficiency and fairness;
- Requirement:
 - Flexible;
 - Scalable;
 - Bounded;
 - Continuative;
 - Metric independent;

Access Grid Integration



Several Interesting Network Services

- Capability matchmaking;
- Multicast monitoring;
 - Hierarchical K-means clustering;
 - Fairness of multicast
- Audio stream transcoding;
- Video stream selection;
- Audio volume balance controller;
- ...

Summary

- Ready:
 - Capability matchmaker;
 - Multicast hierarchical clustering;
- In progress:
 - A study of fairness of multicast;
 - Access Grid integration;
- Planning:
 - Network service design.



Reference

- [1] Gao, H., Papka, M. E., and Stevens, R., The Design of Network services for Advanced Collaborative Environment. In *Proc. 3rd Workshop of Advanced Collaborative Environment*, Seattle, Washington, June 22, 2003.
- [2] Gao, H., Judson, I. R., Uram, T., Disz, T. L., Papka, M. E., Stevens, R., Capability Matching of Data Streams with Network Services. In *4th IEEE/ACM International Symposium on Cluster Computing and the Grid*, Chicago, Illinois, April 19-22, 2004.
- [3] Gao, H., Nickless, W.K., Disz, T.L., Papka, M.E., Stevens, R.L., Topology Analysis for Multicast Grid, Technical Report.
- [4] Gao, H., Papka, M.E., Stenvens, R.L., A study of fairness for mulicast, *in Progress*.
- [5] Codes of projects at
ci-cvs.uchicago.edu:/var/cvsroot/NetworkServices

Credits

This work is supported in part by:

- the Mathematical, Information, and Computational Sciences Division subprogram of the Office of Advanced Scientific Computing Research, Office of Science, U.S. Department of Energy, under Contract W-31-109-ENG-38;
- the National Science Foundation under Grant No. ANI-0222509;
- Microsoft research
- Federal funds from the National Library of Medicine, National Institutes of Health, under Contract No. N01-LM-3-3508 and Grant R01-LM-06756

