The Design of Network Service Architecture and Its Application to AG 2

Retreat 2003

Outline

- Review:
  - Assumptions of AG
  - Goals of AG2
- Novel proposal for Network Service Architecture (NSA)
  - Properties of NSA
  - General approach of NSA
- Current and future work
- Summary

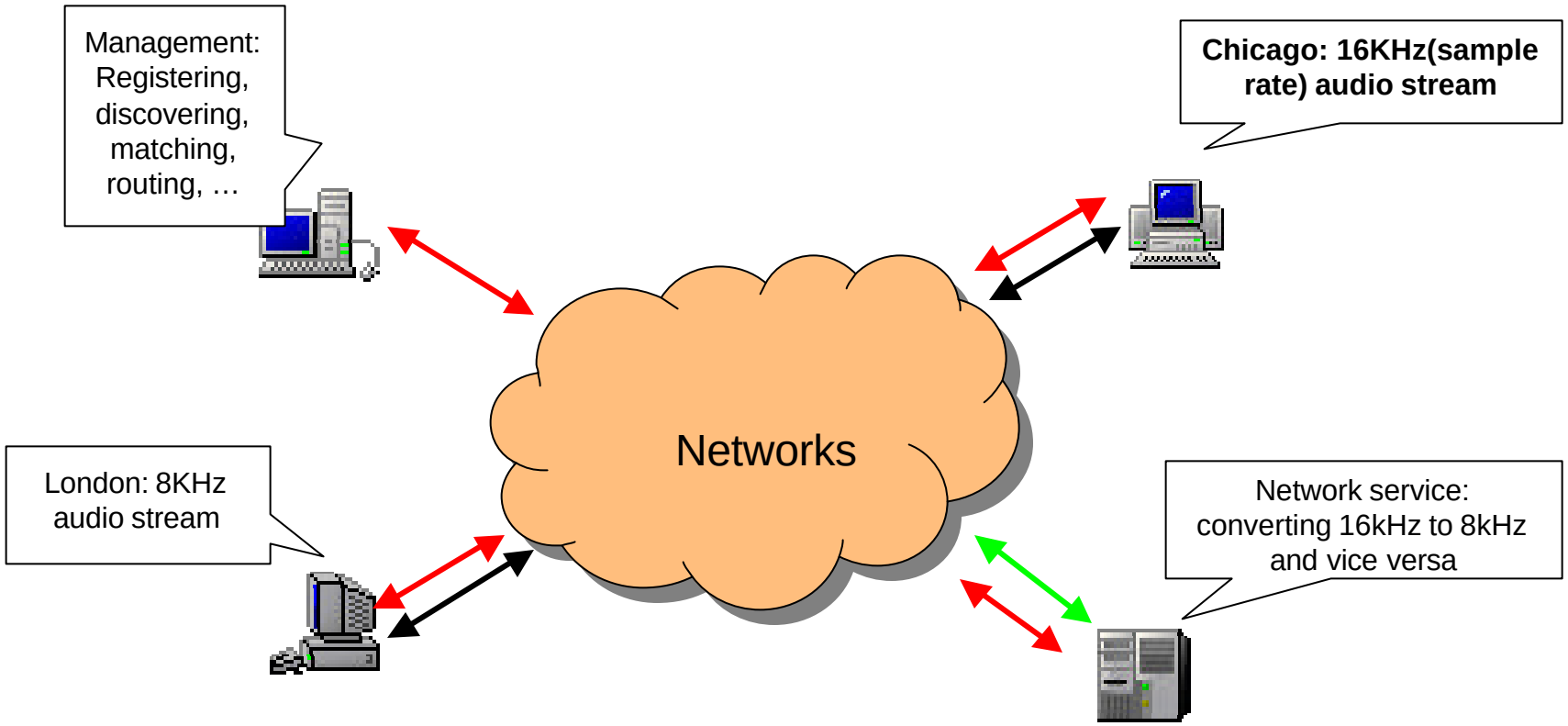
Assumptions of AG

- The original assumption:
  - Higher bandwidth;
  - Higher computational capability;
- Implementation of AG1
  - Location constraints;
  - Resource constraints (bandwidth, computing power, etc.).

Goals of AG2

- Building environment for better remote interactions
  - Free of location constraints
- Relaxing resource constraints
  - Various bandwidth, computing capability
  - Multi-platform support

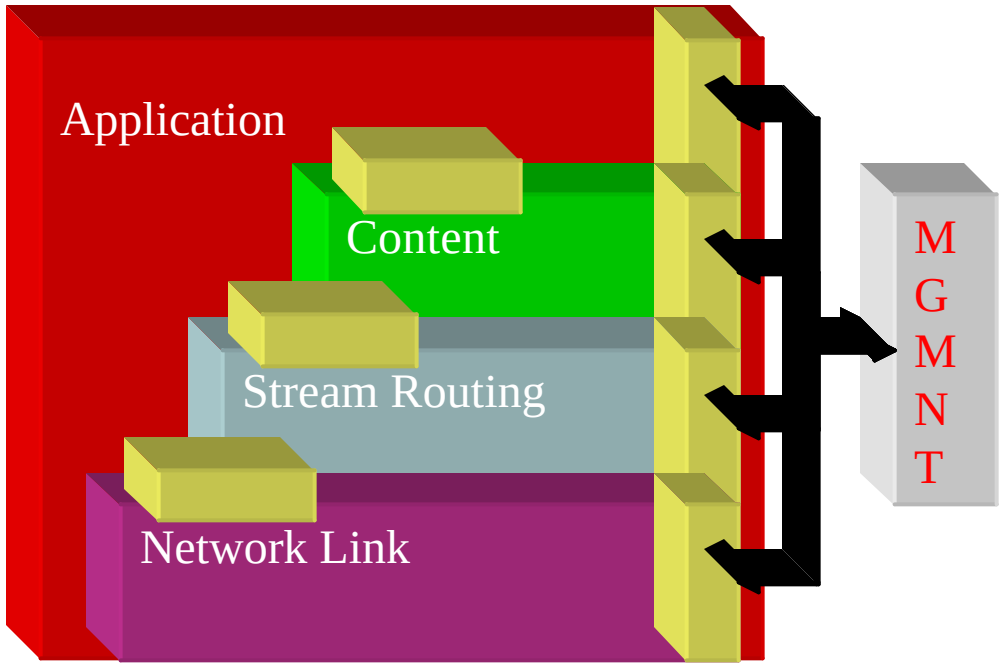
Example


Management: Registering, discovering, matching, routing, …
Chicago: 16KHz(sample rate) audio stream
Networks
London: 8KHz audio stream
Network service: converting 16kHz to 8kHz and vice versa

Properties of Network Service and Network Service Architecture

- Properties of network service
  - Network service as a network resource object
  - Decentralization
  - Heterogeneous conditions
- Properties of network service architecture
  - Soft-state
  - Decentralization
  - Self-adaptable
  - Transparent
  - Stream-Capability

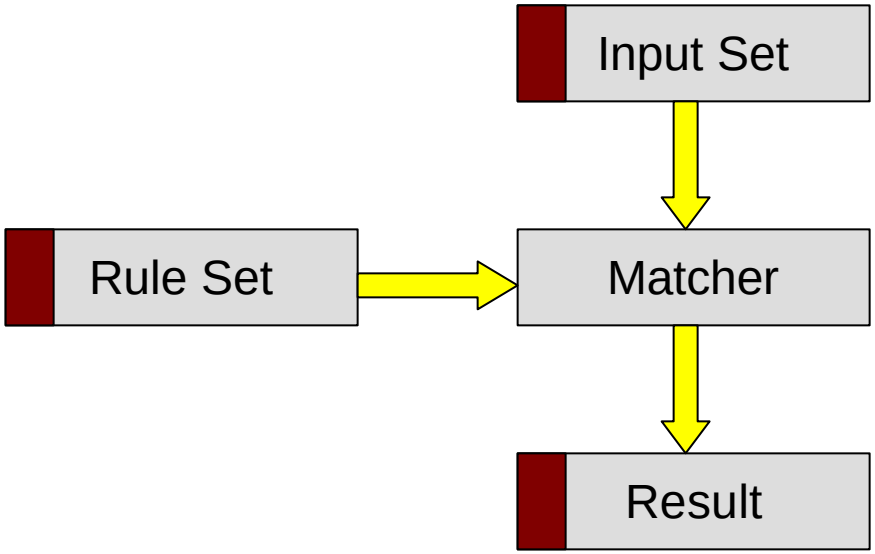
Network Service Architecture



Network Service Architecture

- Applications
  - Resource, networks, …
- Content-Interceptor
  - description files defined by specific schemas
- Stream Route
  - Finding optimal streaming topology
- Network Links
  - Reflector, Bridges, …

General Approach: General Matcher



General Approach: Matching Algorithm

- Capability Space and Capability Vector:

$$\vec{v} = (v_1, v_2, \ldots, v_n) \in R^n$$

- Network Service as Transformation Matrix:

$$\vec{v}_1 = T\vec{v}_2$$

General Approach: Matching Algorithm (cont.)

- Finding the best service in two sets:

$$T_{1,2} = T(S_1^n, S_2^n) = \left[T_{i,j}\right]_{\max\left\|\vec{v}_{1,i}\right\|}, \qquad \text{where } \vec{v}_{1,i} = T_{i,j}\vec{v}_{2,j}$$

- Finding a set of services for multi-sets:

$$S_1^n, S_2^n, \ldots, S_k^n, S_i^n \subseteq R^n, i = 1, \ldots, k.$$

$$\vec{S}_1^n \perp \vec{S}_2^n \perp \ldots \perp \vec{S}_l^n, \; l \le k$$

$$\{T_{1,2}, T_{1,3}, \ldots, T_{i,j}, \ldots, T_{l-1,l}\}, \text{ where } T_{i,j} = T(\vec{S}_i^n, \vec{S}_j^n)$$

General Approach: Stream Topology Algorithm

- Active unconnected session:
- We borrow nodes from non-active session to comprise a connected graph:
- Finding the minimum cost connected graph:

$$G(V, E)$$

$$G''(V'', E'')$$

$$C_{opt} = \min_{V'',E''}(C(G''(V'', E'')))$$

Current and Future Work

- Current work
  - Simulation of a model management module: two inputs → one best resolution choice or a list of choices
- Future work
  - Implementation of general matcher
  - Implementation of stream topology algorithm
  - Integration of network service architecture

Summary

- Novel proposal of Network Service Architecture
  - General Matcher and Matching Algorithms
  - Streaming Topology Algorithms