

RMCP: Remote Music Control Protocol

— Design and Applications —

School of Science and Engineering, Waseda University, Japan

Masataka Goto Ryo Neyama Yoichi Muraoka

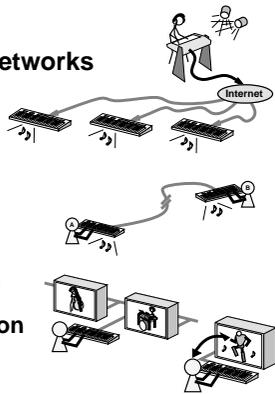
No.1

1997/09/27 ICMC97

1. Introduction

□ Our Goal

- Integrate MIDI and computer networks
- Network musical applications
 - live MIDI broadcasts
 - remote sessions via network
- Computer supported sessions
 - human-computer improvisation
 - interactive-graphics displays



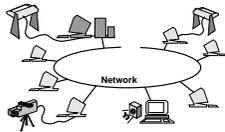
No.2

□ Network Protocol for Musical Information

- Transmit symbolized musical information via network
- Network musical applications
- Distributed implementation of music-related software
 - achieve good load-balancing
 - exploit various facilities
 - connected with different computers



Efficient information sharing over network is important



No.3

□ Related Work

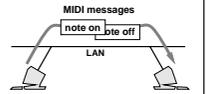
- MIDI-based network protocols
 - connection-oriented
 - not emphasize low-latency information sharing among multiple distributed processes
- MIDI
 - weak in efficient information sharing among devices
 - low bandwidth / just for local communication
- Non-MIDI-based music protocols
 - generally presuppose the use of special devices

No.4

2. RMCP (Remote Music Control Protocol)

□ Network Protocol for Music Applications

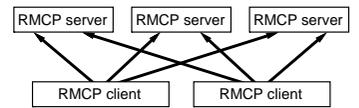
- Share symbolized musical information via networks
 - integrate MIDI and LAN (Ethernet)
- Efficient information sharing among processes
 - connection-less protocol on UDP/IP
 - support broadcast-based information sharing without the overhead of multiple transmission
- RMCP packet
 - include various musical information



No.5

□ Multi-server Multi-client Model

- RMCP clients
 - generate RMCP packets
 - broadcast them to all RMCP servers
 - connection-less one-directional transmission
- RMCP servers
 - receive all the broadcast RMCP packets
 - utilize them in various ways
 - sound output / music visualization / music-driven CG
 - add various servers without any extra packets



No.6

□ Design Policy

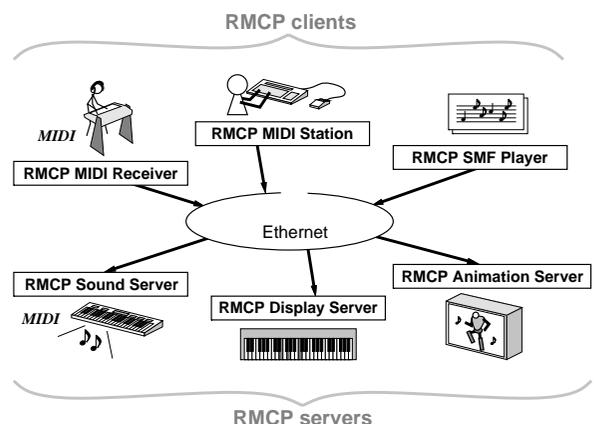
Implement necessary functions as small different processes so that they can be reusable



- Easy system implementation and expansion
 - each process can be devoted to its own small function
 - new function is achieved just by adding a new server
- Good load-balancing
 - easy to allocate RMCP servers and clients on distributed computers

No.7

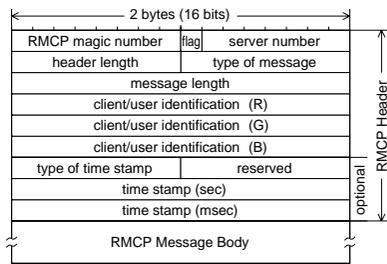
□ Basic RMCP Servers and Clients



No.8

3. Implementation of RMCP

□ RMCP Packet



• Basic message types

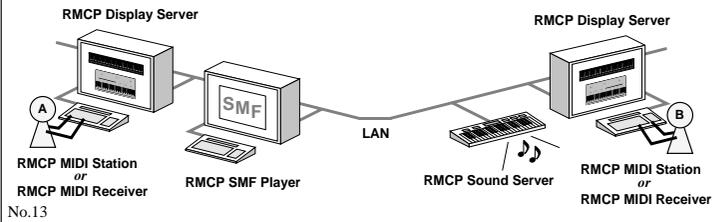
- MIDI information: transmitting MIDI messages
- beat information: beat synchronization
- chord information: chord name and voicing
- animation information: controlling computer graphics

No.9

4. Applications

□ Networked Session

- Several players perform music together via Ethernet
- listen to other players' performances
- see visualized performances



No.13

□ Time Scheduling Using Time Stamps

• Two kinds of RMCP packets

packet with time stamp

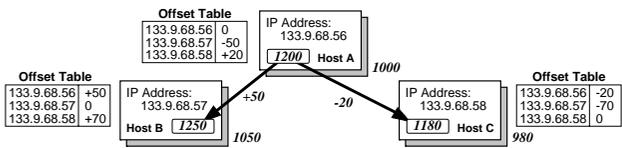
- received before time stamp: processed on time
- received after time stamp: processed immediately

packet without time stamp

processed immediately

• RMCP Time Synchronization Server

enable RMCP servers to handle time stamps as if all the internal clocks were synchronized



No.10

□ Networked Session

No.14

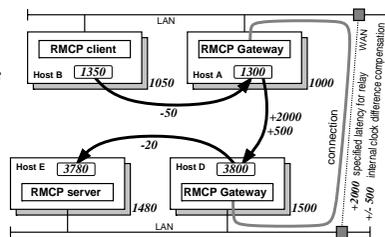
□ RMCP over the Internet

• RMCP Gateway

bidirectional relay of RMCP packets over the Internet
 connection-oriented protocol on TCP/IP
 RMCP servers and clients can communicate as if different LANs were the same network

relay with the specified latency

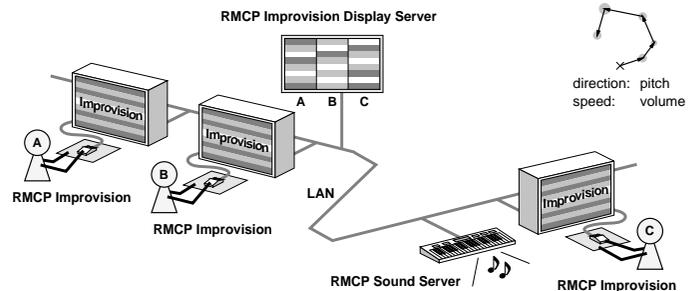
live MIDI broadcasts
 RemoteGIG



No.11

□ Improvisation

- Musical-instrument interface for untrained novices
- improvise unconventional music easily by clicking and dragging a computer mouse



No.15

□ Experimental Results

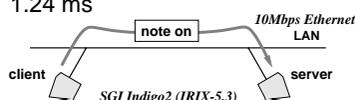
• RMCP programming libraries in both C and Java

tested on various computers and OSs

- IRIX-5.3 IRIX-6.2 Solaris-2.5 SunOS-4.1.3
- HP-UX Linux-2.0 Windows-95 Windows-NT

• Communication delay between server and client

- Ave: 0.30 ms Min: 0.28 ms
- SD: 0.06 ms Max: 1.24 ms



• Fast enough compared with MIDI (31.25 Kbps)

No.12

□ Improvisation

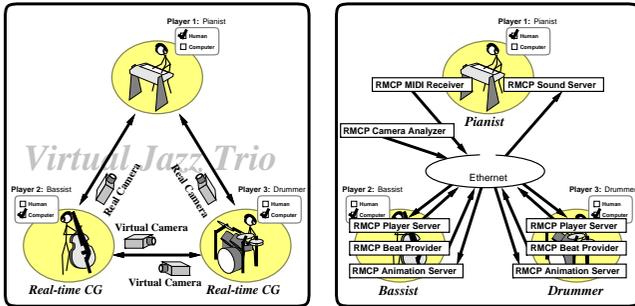
No.16

□ VirJa Session

• Virtual jazz session system

multimodal interaction among all players

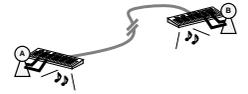
sounds / 3D computer graphics / gestures



No.17

□ RemoteGIG

• Remote session with delay



impossible to avoid network latency over the Internet
models of traditional sessions are not useful

RMCP gateways provide a certain constant latency
with very small deviation

• Innovative remote session over the Internet

overcome long network latency

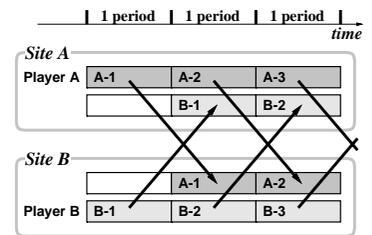
offer a new possibility for future remote sessions

No.21

□ VirJa Session

No.18

□ RemoteGIG



• Assumption

tempo: constant

chord progression: repetitive (ex. 12-bar blues)

playing background drums to keep the tempo

• Turn the network latency to its advantage

other players' performances are delayed

for the constant period of chord progression

delayed performances can fit the chords

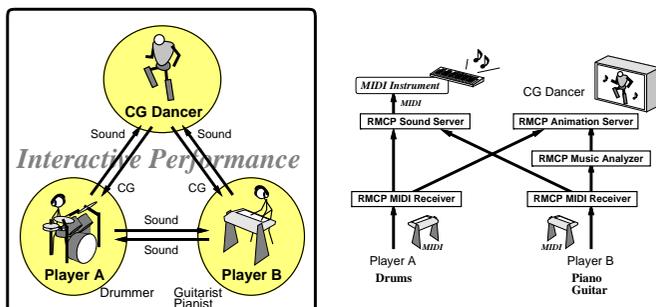
No.22

□ Virtual Dancer "Cindy"

• Interactive performance of a music-controlled CG dancer

two players choreograph Cindy by their improvisation

interact through music and 3D computer animation



No.19

□ Virtual Dancer "Cindy"

No.20

5. Conclusion

□ Summary

• RMCP: Network protocol for music applications

• Multiple distributed processes can share symbolized musical information such as MIDI

• Features:

efficient broadcast-based information sharing over LAN

time-scheduling using time stamps

live MIDI transmission over WAN such as Internet

• Utilized for various applications

No.23

□ Future Work

• Provide an API "Jam" to make

RMCP and MIDI usable in Java applets

• Implement various RMCP-based applications

□ RMCP Software Package Distribution

<http://www.info.waseda.ac.jp/muraoka/members/goto/RMCP/>

basic RMCP server and client programs

RMCP programming libraries

No.24