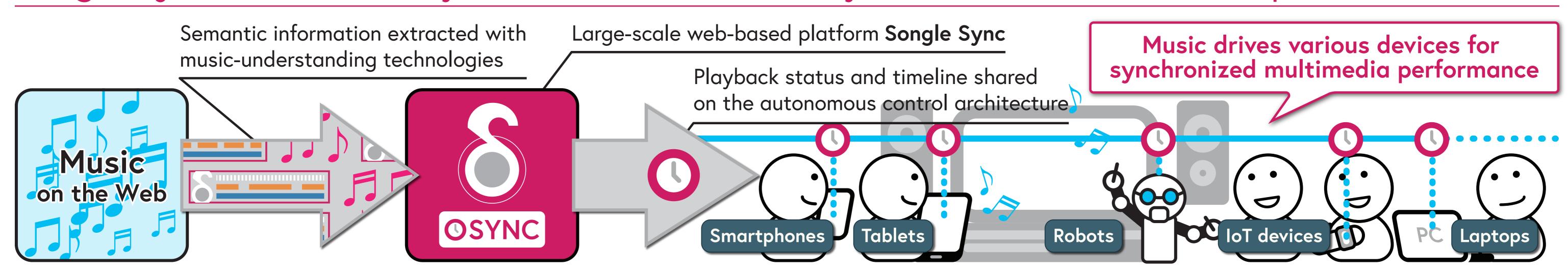
# Songle Sync: A Large-Scale Web-based Platform for Controlling Various Devices in Synchronization with Music

Jun Kato, Masa Ogata, Takahiro Inoue, Masataka Goto National Institute of Advanced Industrial Science and Technology (AIST)

ACM Multimedia 2018 Oct. 25, Poster Session 5

## Songle Sync drives a variety of hundreds of devices synchronized with a musical piece on the web



## Music-Driven Multimedia Performance at Scale

Bring-Your-Own-Device (BYOD) experience for smartphones!



Automatic music analysis has enabled

music-driven multimedia performances



Enjoy the synchronized performance!

Our work is the first attempt to provide a platform on which music-driven performances involve a variety of hundreds of devices.

## **Dynamic** Hardware Setup

Various Internet-connected smartphones can join the session

### Scalable Control of Devices

synchronizes >1000 devices without significant latency nor jitter

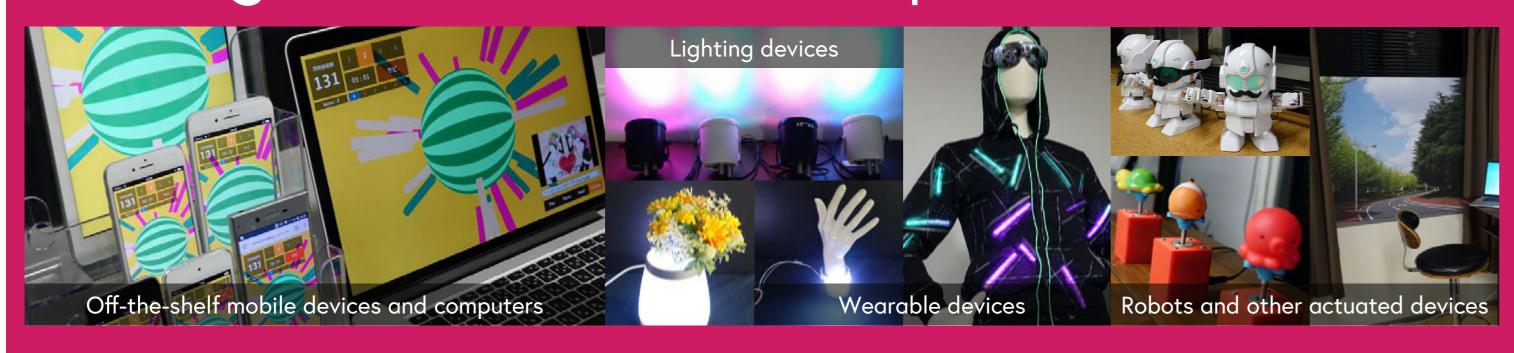


## Stable Control of Devices

works stably under challenging networking environments

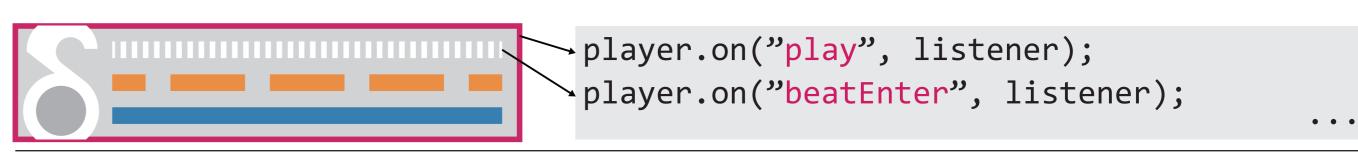


## Heterogeneous Hardware Setup



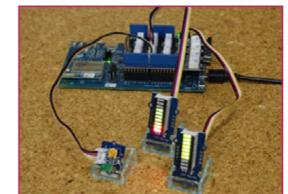
Various JavaScript-driven devices (e.g., Raspberry Pi and Intel Edison) can be controlled since standard web technologies are utilized

## Open Platform with Development Kit



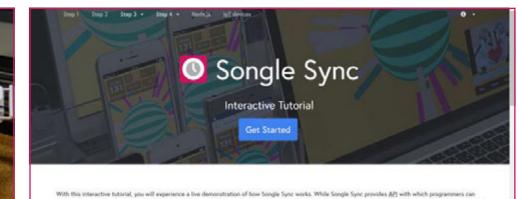
Event-driven APIs for easily controlling devices with JavaScript

- Code for one device can synchronize hundreds of devices
- No need to worry about networking and synchronization



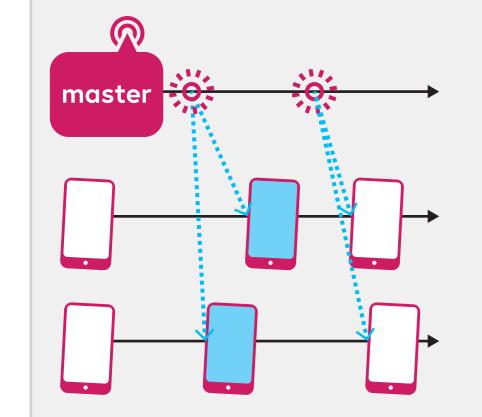






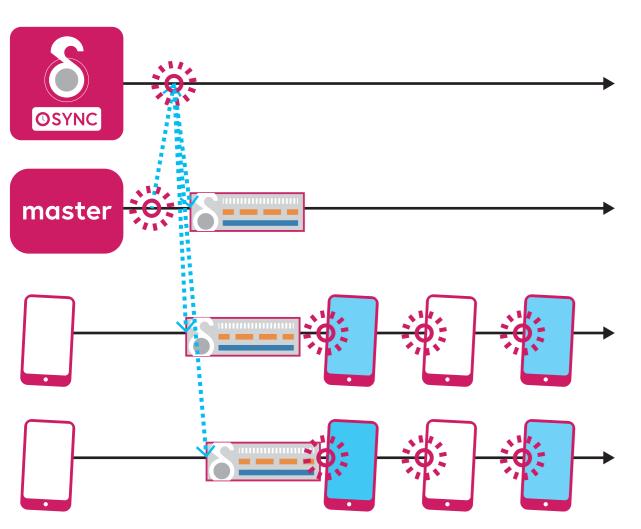
Example programs and interactive tutorials

## Novel "Autonomous Control Architecture"



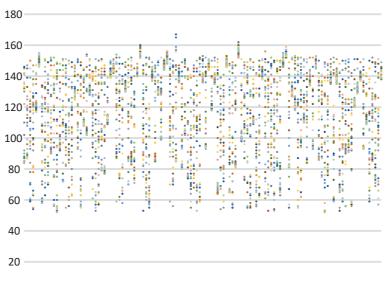
#### Conventional "Always-On Architecture"

- Widely used in live performances with dedicated hardware/smartphone apps
- Master node emits a command at every event
- Slave nodes react to the command
- Per-event communication → latency and jitter



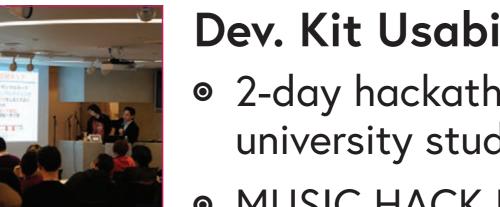
- Master node chooses a musical piece
- Songle Sync distributes timings of musical elements (e.g., beats and chords)
- Playing status is shared periodically
- Every node knows event timings before the multimedia performance
- No per-event communication → theoretically no latency nor jitter
- NTP-like protocol to sync clocks

## Evaluations



### Performance Comparison

- Always-On: **0-180**[ms] delays
- Autonomous: **-20-0**[ms] delays
- - <100[ms] jitter observed by a 960fps camera



- Dev. Kit Usability Test
- 2-day hackathon with 24 university students
- MUSIC HACK DAY Tokyo

### Deployments in the Wild





- hardware setup Live performance with >275 synchronized smartphones
- Recent event synchronized **>1200** devices