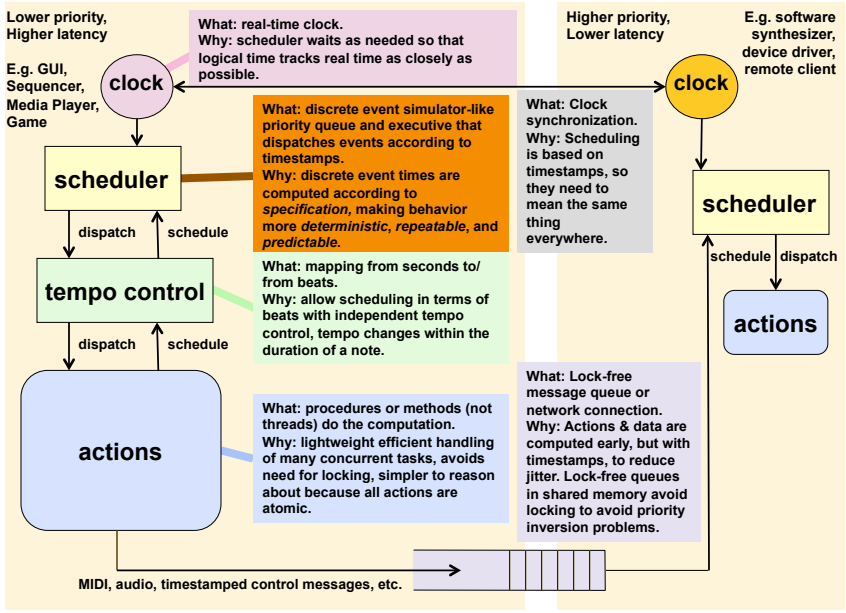
Week 15 – Semester Review

Roger B. Dannenberg
Professor of Computer Science, Art & Music
Carnegie Mellon University

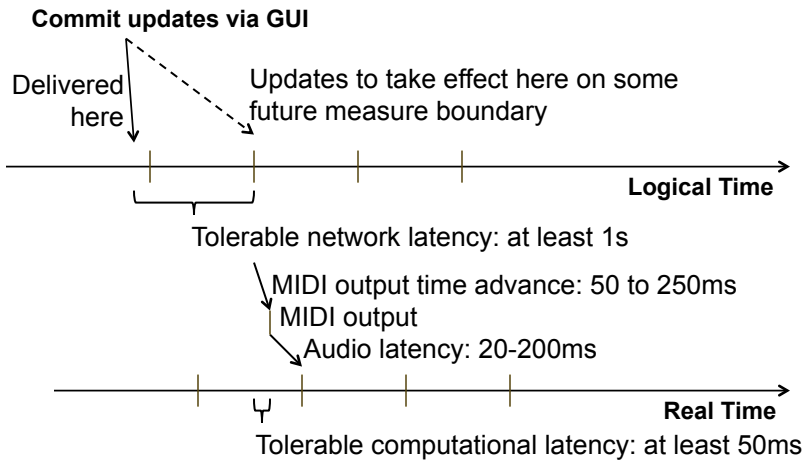
What Did You Learn?

- (i.e. what do I think I taught you…)
- Hint: in the syllabus, I listed concepts with each week.

Grand Guide to Interactive Real-Time Systems



Concert System Timing



Week 1 – Midi Standard

- Notes, note-on, note-off
- Channels in MIDI
- Control change and what control changes affect
- timing in MIDI and MIDI files
- tracks in MIDI files

Week 2 – Discrete Event Simulation

- Why talk about simulation?
  1. Precise timing: you can compute ideal execution times and avoid drift

```
while (true)                    vs      time = 0
    play_a_sound()                      while (true)
    wait(1 second)                          play_a_sound()
                                            time = time + 1
                                            wait_until(time)
```

  2. Many interleaved tasks without threads
     - Efficiency
       - no stacks
       - simple context switch
     - No need for locks, synchronization

Week 2 – Scheduling

- abstract interface
- linked list
- sorted list
- heapsort
- time wheel *or calendar queue*
- time wheel + heapsort

Week 3 – Forward Synchronous Systems

- logical time as a specification of desired behavior
- tempo as slope of time map
- time as integral of tempo
- beats as integral of 1/tempo
- tempo and control parameters through computation in Formula
- nested and multiple tempo
- forward synchronous systems
- event buffers
- **how do event buffers reduce jitter at the cost of latency?**
- examples of event buffering in applications, device drivers, hardware.

Week 4 - Music Theory

- Pitches, scales, notation
- Duration, measures, time signatures, notation
- Intervals, chords
- Form: repetition, variation

Week 5 – Music Generation, Algorithmic Composition

- music as time series data
- Markov models
  - limited local context
  - estimating transition probabilities
- Pattern Generators
- music as formal language
- hierarchical structure and its relationship to grammars
- Role of suffix trees

Week 6 – Networking, Clock Synchronization

- network latency
- OSC & O2
  - addressing mechanisms
  - timing mechanisms
  - network addressing
  - reply, status, acknowledgements (or lack thereof)
- Clock drift
- Clock skew

Week 7 – Music Understanding and Sequencers/MIDI Files

- Computer Accompaniment
- Style Recognition
- Audio Alignment

- MIDI data
- MIDI meta-data: titles, lyrics, time signature,…
- Delta times with variable-length encoding
- Tracks

Week 9 – Audio Architectures

- Samples, frames, blocks
- audio signal processing computation organization –
  - blocks for efficiency, graphs of unit generators
- why must audio processing be synchronous?
- callback/asynchronous API– thread is provided, asynchronous
- blocking/synchronous API– user controls thread and synchronization
- typical scheduling strategies for audio applications –
  - Understand need for buffers
  - Buffer size determines latency
- what is the architecture of plug-ins? –
  - unit generator + general API

Week 10 – Web Audio and Concurrency

- Web Audio ≈ unit gens + timed updates + javascript
- locks, semaphores
- Synchronization with message passing
- Real-time issues: blocking, priority inversion
- Lock-free synchronization
- Scheduling:
  - Earliest deadline first
  - Fixed priority

Week 9 – Music Representation

- note lists
- special purpose vs. general/extensible representations
- why is music notation difficult?
- how much notation information is in a MIDI file?
- scores as data types - operations on scores
- hierarchy in music data

Week 12 – Music Robots, Max Family of Languages

- See Pd documentation

Week 13 – Audio Editors

- In place (destructive) editors
- Non-destructive editors
- Audacity sequence data structure:
  - O(1) disk operations for delete, cut, copy, paste, undo, redo

Week 12 – Query By Humming and Music Fingerprinting

- Query By Humming - what it does
- DP for approximate substring matching
  - Interval matching is transposition invariant
  - IOI-ratios capture rhythm information
- DTW – sample pitch contour at equal time intervals
- music fingerprinting
  - What it does – identify exact audio source from specific acoustic features
  - Features
  - Hashing techniques

Week 12 – Music Classification

- music features – e.g. note density, interval size, duration, …
- estimating means
- Bayes Theorem
  - Training data tells you class→feature,
  - use Bayes to get feature→class
- maximum likelihood (of class membership)
- Naïve Bayesian Classifier (assumes Gaussian, simple)

Not on the Exam…

- Roger Linn – authority on sensors and new instruments

Reminders, Discussion

- Course evaluation