Embedded and real time architectures

48433 Software Architecture

What is different

- Why do real time architectures require special consideration?
- What are some characteristics of real time systems that affect the architecture? (10 minutes)
-
- Response time requirements are set by external events or controls
- Usually means that there can be no virtual memory or paging.
- The system must be exhaustively determinate (responses to all circumstances are anticipated)

What is different?

- Why do embedded systems require special consideration?
- What are some characteristics of Internet of Things (IoT) systems that affect the architecture (10 mins)
-
- Limited power, limited memory, limited cpu
- Cost of the hardware and software is constrained. e.g. a smart water meter or electricity meter must be affordable.

Responding to an unknown expansion

- IoT systems usually face an exponential growth in endpoint devices, usually sensors or actuators.
- What architecture techniques would you use to accommodate such an unknown growth? (10 mins)

Interact with distributed through a message bus or highly scalable network.

Isolate the device through a facade. That enables the device to change the way it implements its features while retaining the same interaction with its controlling systems

Responding to an unknown future

- IoT systems are likely to last a long time. Think in terms of the electricity distribution system or water distribution. Over their lifetime connected sensors, actuators and controllers are likely to evolve.
- What architecture techniques could you use to support constantly changing sensors, actuators and controllers? (10 mins)

Communicate through facades.

When it absolutely must be there

- Real time control systems must respond to events within time limits. Aside from using a really fast CPU, what architecture techniques can be used to ensure that responses are within allowable time limits?

Use a real time operating system (no paging, no virtual memory)
No indeterminate waiting on semaphores or events

What if it fails?

- Computers and operating systems are quite reliable now. Is there really a need for failover systems? What techniques do high reliability systems employ to ensure high reliability?

- Data errors can cause different results. So, run the same algorithm through an odd number of processes, with a voting system to determine which result to accept.
- If a process fails, kill the process and start a new instance. This requires process to restart whatever the failed process was doing.
- A recommendation from a very experienced high reliability system developer was to not spend time determining what went wrong. Just kill the process and restart it. Fault diagnosis should be offline, based on logged data.