Real-Time

versus

Non Real-Time Operating Systems
Real-Time Applications

- A “real-time” application is one in which the passage of time is important.
  - The application must perform processing at a rate that is comparable to the rate at which the environment evolves.
  - Processing tends to be depend on the rate at which IO operates:
    - GPS Receivers, CD/MP3 Players, Video, etc.
Real-Time Applications

Real-time applications, continued:

- Time-critical tasks – At least some of the tasks in a real-time application are linked to time:
  - “hard real-time” – Strict time deadlines. If the program missed a deadline, the system is broken.
  - “soft real-time” – Less strict deadlines. It may be acceptable to occasionally miss a deadline.

- Periodic tasks – Tasks often run in a predictable pattern.
Real-Time Applications

- Real-time applications, continued:
  - Task synchronization – Execution of tasks is often highly correlated in time and with the evolution of the environment.
  - Task communications – Tasks may tend to trigger other tasks. There may be substantial data that “flows” between tasks.
Inter-Task Data Flow

Source: Gomaa, “A software design method for real-time systems.”
Flow Control of Tasks

Source: Gomaa, “A software design method for real-time systems.”
Scheduling in a Real-Time System

- The most obvious thing to manage in a real-time system is time by scheduling tasks. However, the scheduling problem has many dimensions:
  - Task execution – when should a task be scheduled.
  - Power – Some systems save power by dynamically varying voltage.
  - Data – Varying how/when data items are transferred.
- These issues all inter-relate.
Each task may have several characteristics that the real-time system must account for:

- Execution time – how much time does the task require (minimum and maximum)?
- Atomicity – can the task be divided over time?
- Deadline – When must the task complete in order for the results to be useful?
- Latency – How much latency and variability?
There are several ways tasks may be scheduled:

- **Time Driven** – The start and stop times of each task are determined *a priori* and are strictly managed by the scheduler.
- **Priority Driven** – Each task is given a priority.
  - EDF – Earliest Deadline First
  - RM – Rate Monotonic, priority based on period
- **Share Driven** – Based on “share” of resources.
A Hybrid Scheduler

Source: Tan, “A Hybrid Scheduling Scheme for Hard, Soft and Non-Real-Time Tasks.”
Missing Deadlines

Source: Tan, “A Hybrid Scheduling Scheme for Hard, Soft and Non-Real-Time Tasks.”