Task Model

謝仁偉 助理教授
jenwei@mail.ntust.edu.tw
國立台灣科技大學 資訊工程系
2008 Fall

Major References:
Real-Time Computing，國立交通大學，張立平教授

Terminologies (1/2)

- Job: a unit of work that is scheduled and executed by the system.
- Task: a set of related jobs which jointly provide some system function.
- The release time of a job: the instant of time at which the job becomes available for execution.
- Response time: the length of time from the release time of the job to the instant when it completes.

Terminologies (2/2)

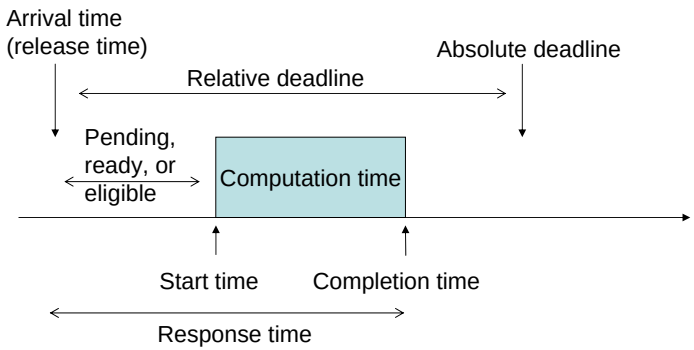
- The deadline of a job: the instant of time by which its execution is required to be completed.
  - Relative deadline: the maximum allowable response time of a job.
  - Absolute deadline: release time + relative deadline
- Scheduler: a scheduler is a module that allocates processors and resources to jobs and tasks.
- Schedule: an assignment of all the jobs in the system on the available processors produced by the scheduler.

Feasible vs. Schedulable

- A valid schedule is a feasible schedule if every job completes by its deadline (or, in general, meets its timing constraints).
- A set of jobs is schedulable according to a scheduling algorithms if when using the algorithm the scheduler always produces a feasible schedule.

System Model

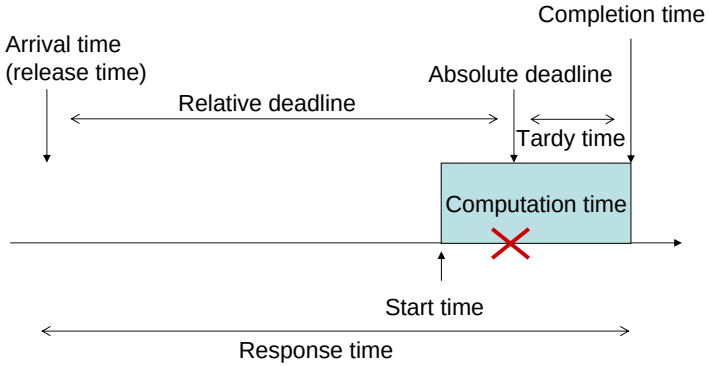
- A job with real-time constraints



The job completes before its deadline, that means the deadline is satisfied.

System Model

- A job with real-time constraints



The job completes after its deadline, that means the deadline is violated or an overflow occurs.

Periodic Task Model (1/2)

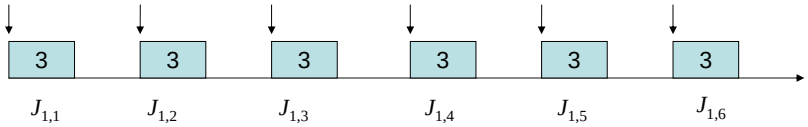
- Each periodic task, denoted by T_i, is a sequence of jobs.
  - The period (p_i) of the periodic task T_i is the minimum length of all time intervals between release times of consecutive jobs in T_i.
  - The execution time (e_i) is the maximum execution time of all the jobs in it.
  - The release time $r_{i,1}$ of the first job $J_{i,1}$ in each task T_i is called the phase (ϕ_i) of T_i.
  - A job in T_i that is released at t must complete D_i units of time after t; D_i is the relative deadline of the task T_i.

Periodic Task Model (2/2)

- Priority
  - Reflect the urgency of jobs
  - Any job inherits its task's priority
- Preemptivity
  - As a high-priority task arrives, it preempts the execution of any low-priority tasks

A Purely Periodic Task

- Jobs of a task T recur every fixed time interval p
- A job must be completed before the next job arrives
  - Relative deadlines for jobs are, implicitly, the period
- T is defined as (p, c)



Periodic task $T_1=(6, 3)$

Hard/Soft/Firm Real-Time Systems (1/2)

- Hard Real Time
  - Hard real-time systems require a guarantee that *all* processing is completed within a given time constraint *every time*.
  - A late response may result in catastrophic consequences.
  - Examples of hard real-time systems include nuclear power plants and avionics control systems.
- Soft Real-Time
  - Soft real-time systems have a less rigorous notion of temporal correctness and the consequences of a late response are not catastrophic.
  - Examples of soft real-time systems include telephone switches and electronic games.

Hard/Soft/Firm Real-Time Systems (2/2)

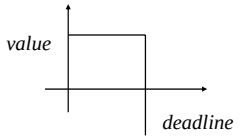
- Firm Real Time
  - Firm real-time has been used to describe applications that require deterministic performance but not hard guarantees of performance.
  - Examples of firm real-time systems include video conferencing systems and network servers such as Web Service Providers (WSPs), Application Service Providers (ASPs), and those that support e-commerce.

http://csce.unl.edu/~goddard/ResearchProjects/VariableRateResourceAllocation.html

Types of Real-Time Systems

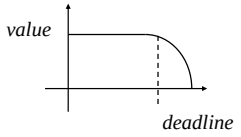
- Hard real-time

  No deadline violation



- Soft real-time

  Low miss ratio or average/worst-case response time



- Firm real-time

  No value after deadlines expire