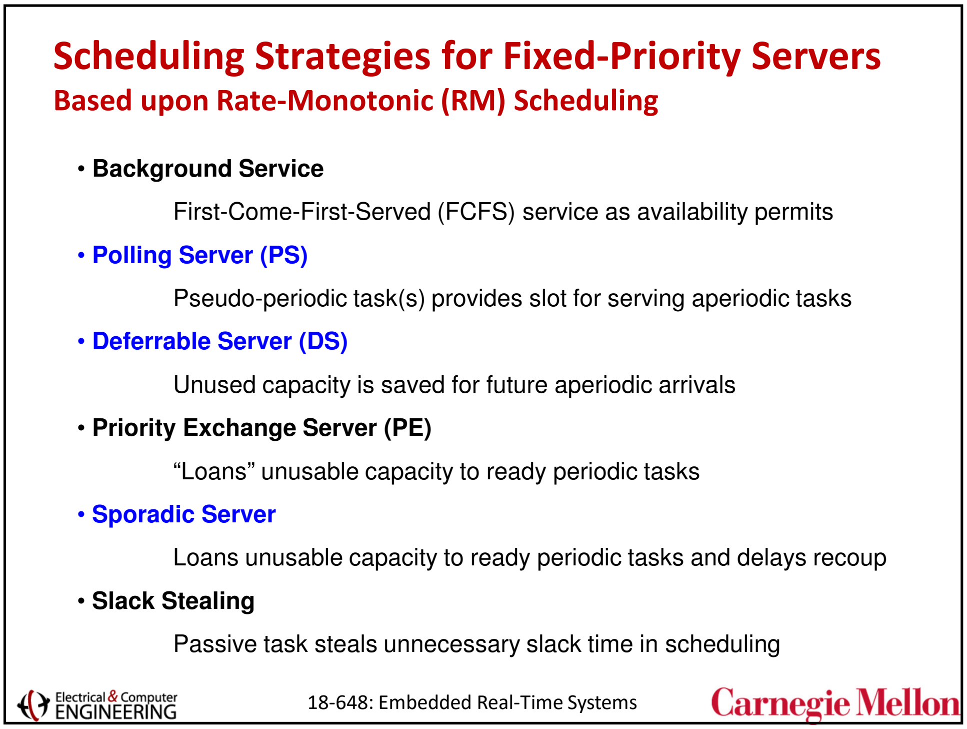
Dynamic-Priority Aperiodic Servers

Raj Rajkumar

Lecture #9

Outline

- Dynamic Priority Schemes for Aperiodic Servers
  - Dynamic Priority Exchange Server (DPE)
  - Dynamic Sporadic Server (DSS)
  - Total Bandwidth Server (TBS)
  - Earliest Deadline Late Server (EDL)
  - Improved Dynamic Priority Exchange Server (IPE)
  - Constant Bandwidth Server (CBS)



Scheduling Strategies for Fixed-Priority Servers

Based upon Rate-Monotonic (RM) Scheduling

- **Background Service**

  First-Come-First-Served (FCFS) service as availability permits

- **Polling Server (PS)**

  Pseudo-periodic task(s) provides slot for serving aperiodic tasks

- **Deferrable Server (DS)**

  Unused capacity is saved for future aperiodic arrivals

- **Priority Exchange Server (PE)**

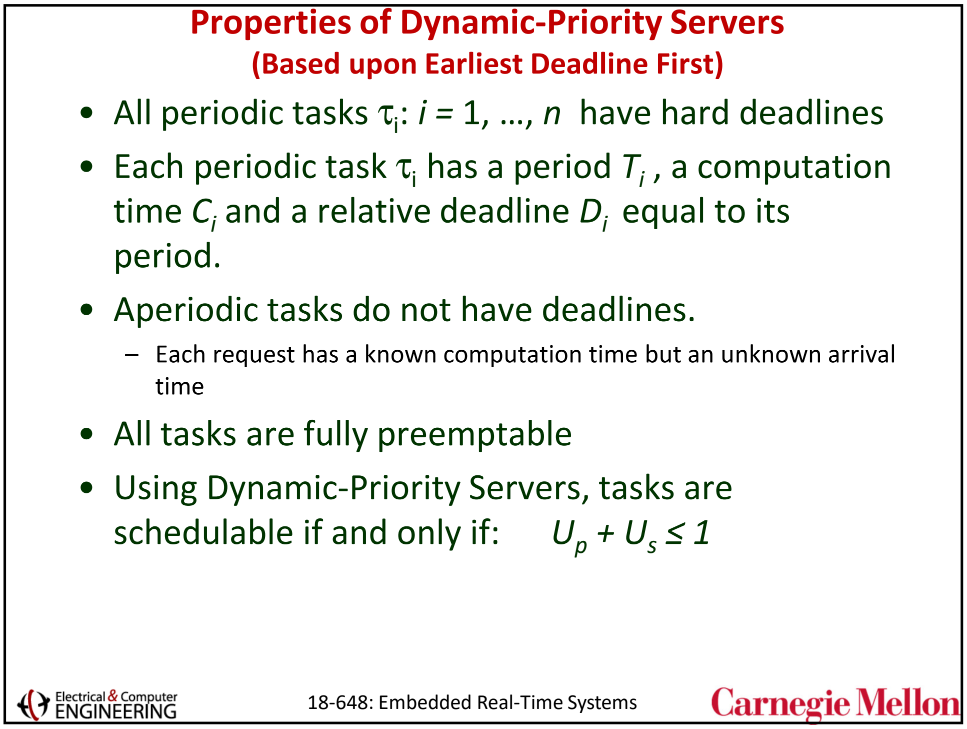
  “Loans” unusable capacity to ready periodic tasks

- **Sporadic Server**

  Loans unusable capacity to ready periodic tasks and delays recoup

- **Slack Stealing**

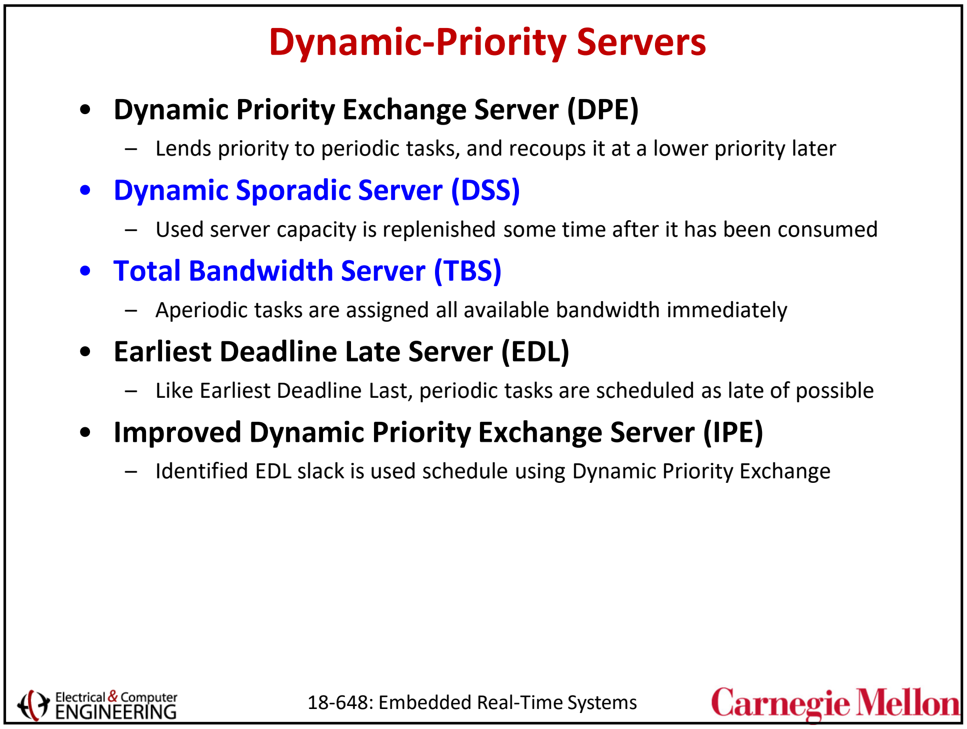
  Passive task steals unnecessary slack time in scheduling



Properties of Dynamic-Priority Servers

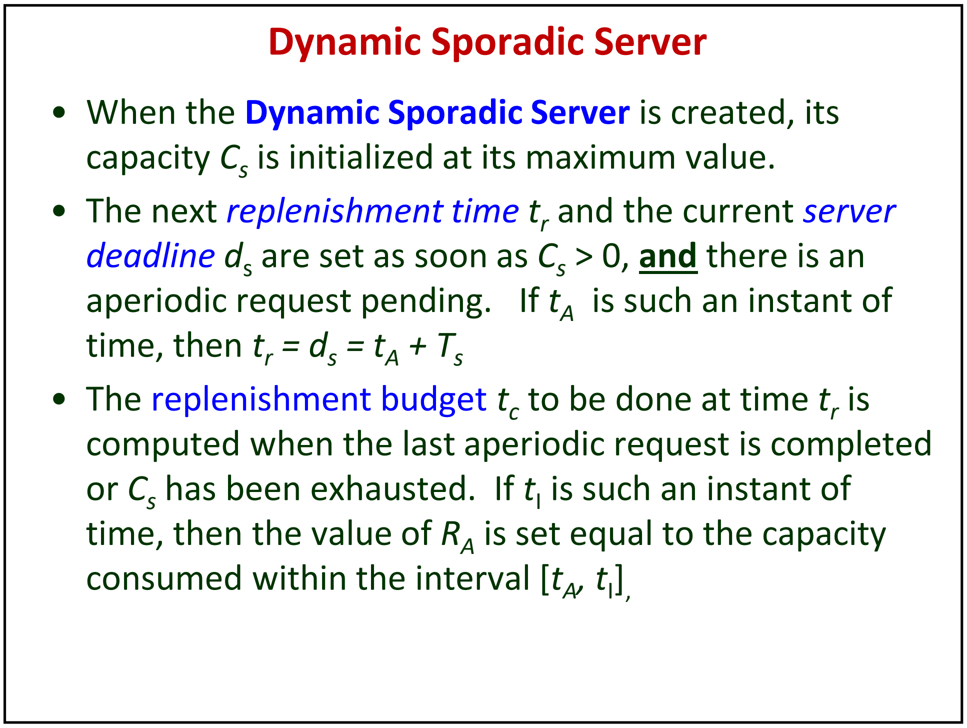
(Based upon Earliest Deadline First)

- All periodic tasks τ_i: $i = 1, \ldots, n$ have hard deadlines
- Each periodic task τ_i has a period T_i , a computation time C_i and a relative deadline D_i equal to its period.
- Aperiodic tasks do not have deadlines.
  - Each request has a known computation time but an unknown arrival time
- All tasks are fully preemptable
- Using Dynamic-Priority Servers, tasks are schedulable if and only if: $U_p + U_s \leq 1$



Dynamic-Priority Servers

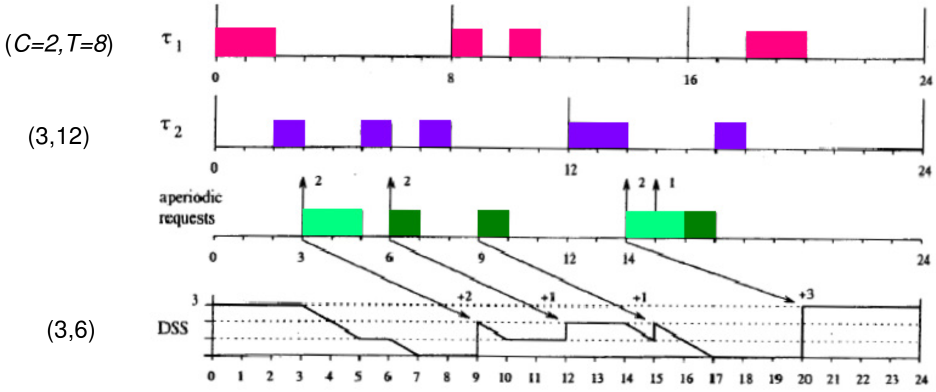
- **Dynamic Priority Exchange Server (DPE)**
  - Lends priority to periodic tasks, and recoups it at a lower priority later
- **Dynamic Sporadic Server (DSS)**
  - Used server capacity is replenished some time after it has been consumed
- **Total Bandwidth Server (TBS)**
  - Aperiodic tasks are assigned all available bandwidth immediately
- **Earliest Deadline Late Server (EDL)**
  - Like Earliest Deadline Last, periodic tasks are scheduled as late of possible
- **Improved Dynamic Priority Exchange Server (IPE)**
  - Identified EDL slack is used schedule using Dynamic Priority Exchange



Dynamic Sporadic Server

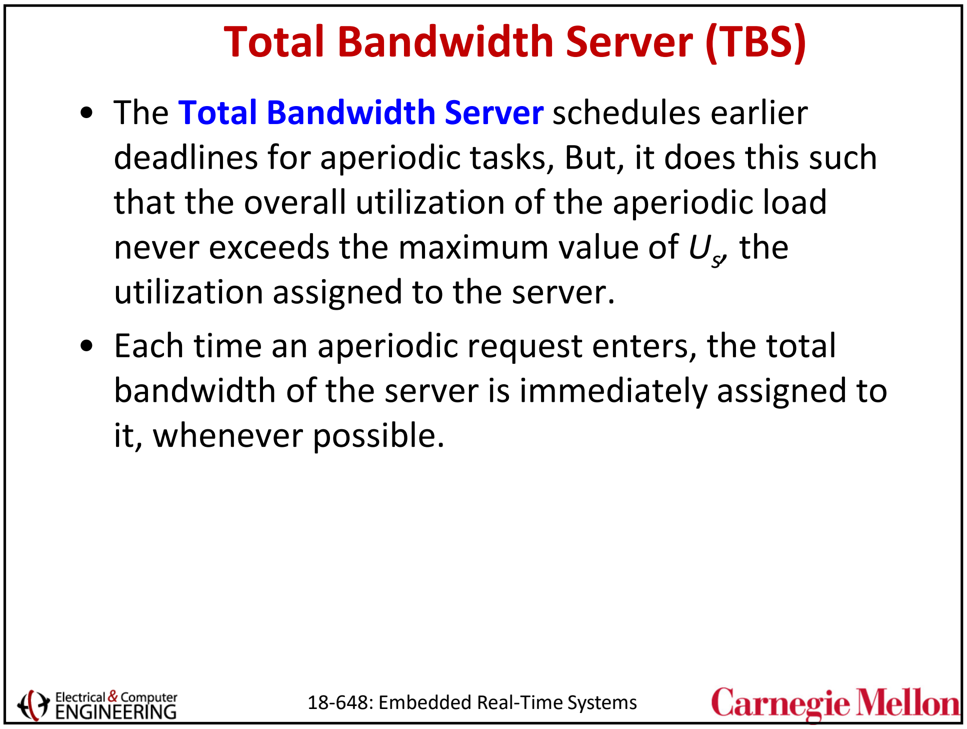
- When the **Dynamic Sporadic Server** is created, its capacity C_s is initialized at its maximum value.
- The next *replenishment time* t_r and the current *server deadline* d_s are set as soon as $C_s > 0$, **and** there is an aperiodic request pending. If t_A is such an instant of time, then $t_r = d_s = t_A + T_s$
- The replenishment budget t_c to be done at time t_r is computed when the last aperiodic request is completed or C_s has been exhausted. If t_I is such an instant of time, then the value of R_A is set equal to the capacity consumed within the interval $[t_A, t_I]$,

Dynamic Sporadic Server Example



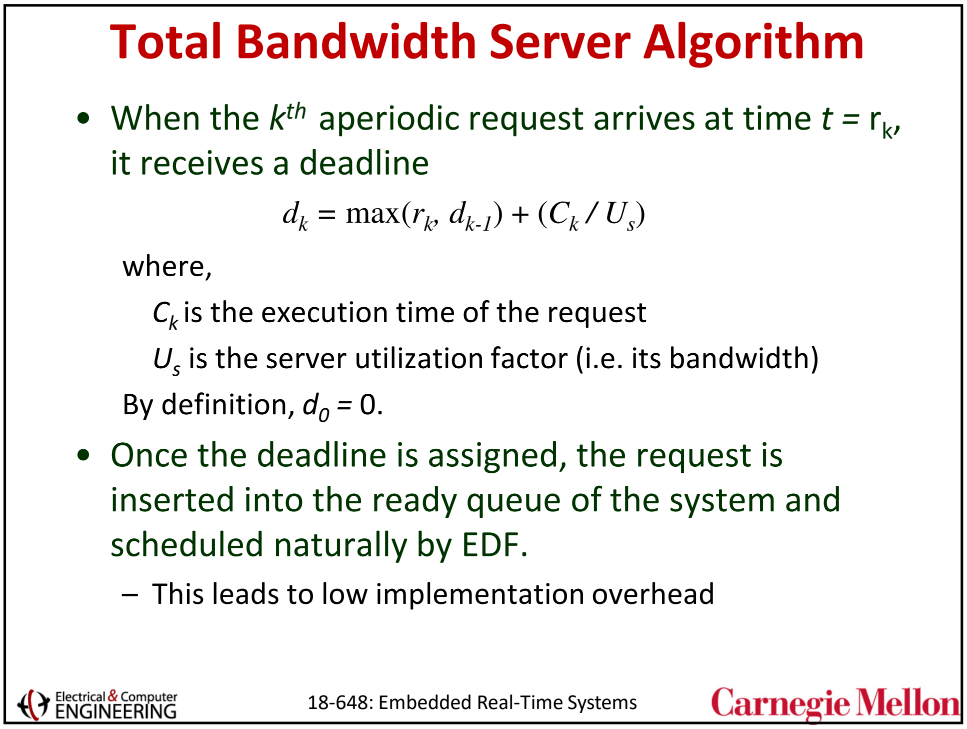
Notes on Dynamic Sporadic Server

- The **Dynamic Sporadic Server** always schedules with a relatively far deadline. Therefore, the response of aperiodic tasks may be relatively long.
- One solution is to shorten the period of the Dynamic Sporadic Server, at the expense of more context switches and lower DSS utilization.



Total Bandwidth Server (TBS)

- The **Total Bandwidth Server** schedules earlier deadlines for aperiodic tasks, But, it does this such that the overall utilization of the aperiodic load never exceeds the maximum value of U_s, the utilization assigned to the server.
- Each time an aperiodic request enters, the total bandwidth of the server is immediately assigned to it, whenever possible.



Total Bandwidth Server Algorithm

- When the k^{th} aperiodic request arrives at time $t = \mathrm{r}_k$, it receives a deadline

$$d_k = \max(r_k, d_{k-1}) + (C_k / U_s)$$

where,

C_k is the execution time of the request

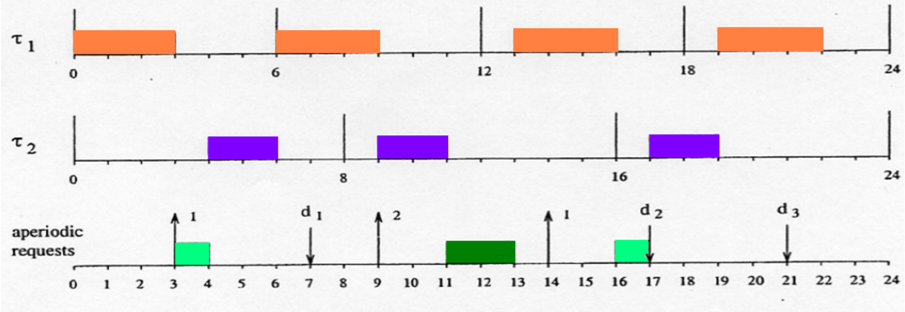
U_s is the server utilization factor (i.e. its bandwidth)

By definition, $d_0 = 0$.

- Once the deadline is assigned, the request is inserted into the ready queue of the system and scheduled naturally by EDF.
  - This leads to low implementation overhead

Total Bandwidth Server Example

$\tau_1 = (3,6) \rightarrow U_1 = 0.5; \quad \tau_2 = (2,8) \rightarrow U_2 = 0.25; U_s = 0.25$



$d_1 = \max(3, 0) + (1/0.25) = 3+4 = 7$

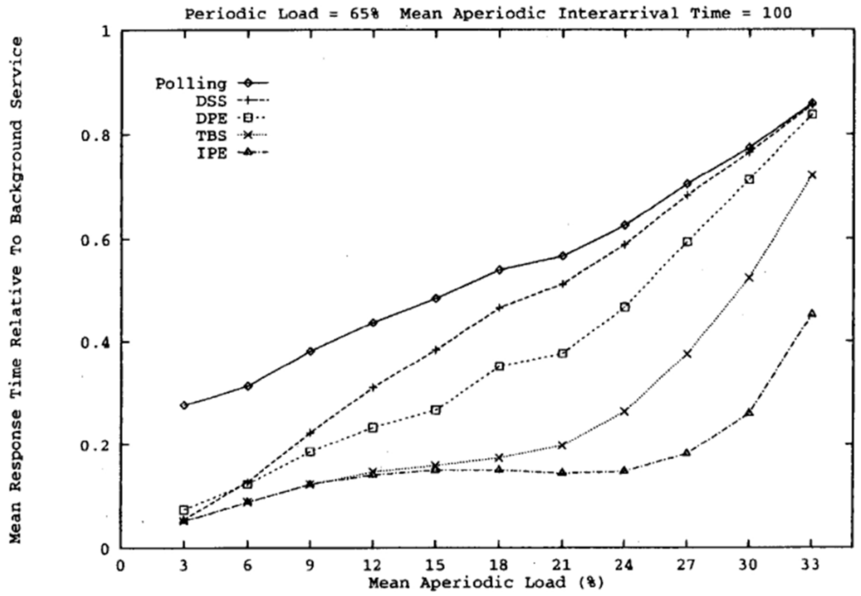
$d_2 = \max(9, 7) + (2/0.25) = 9+8 = 17$

$d_3 = \max(14, 17) + (1/0.25) = 17+4 = 21$

The Constant Bandwidth Server (CBS)

- Server has a maximum budget C_s and a period T_s ($U_s = C_s / T_s$)
- The server is said to be **active** if jobs are pending, otherwise it is idle
- When an aperiodic job arrives, it inherits the server deadline, d_s
- When an aperiodic job executes, the server budget is decreased by the same amount
- When the budget becomes zero, it is recharged to C_s and deadline d_s is increased by T_s
- When a job arrives at time *t* and the server is idle,
  - If remaining budget > $(d_s - t)\ U_s$, the deadline is advanced to $t + T_s$ and the budget is replenished to a maximum of C_s.
  - Else, the job is served with the last server deadline d_s using the current budget
- Jobs are <u>always</u> serviced with the server deadline set at that point

Performance Evaluation of Dynamic Server Schemes



Summary

- Dynamic Priority Schemes for Aperiodic Servers
  - Dynamic Priority Exchange Server (DPE)
  - Dynamic Sporadic Server (DSS)
  - Total Bandwidth Server (TBS)
  - Earliest Deadline Late Server (EDL)
  - Improved Dynamic Priority Exchange Server (IPE)
  - Constant Bandwidth Server (CBS)

Reference

Other EDF-Based SERVERS

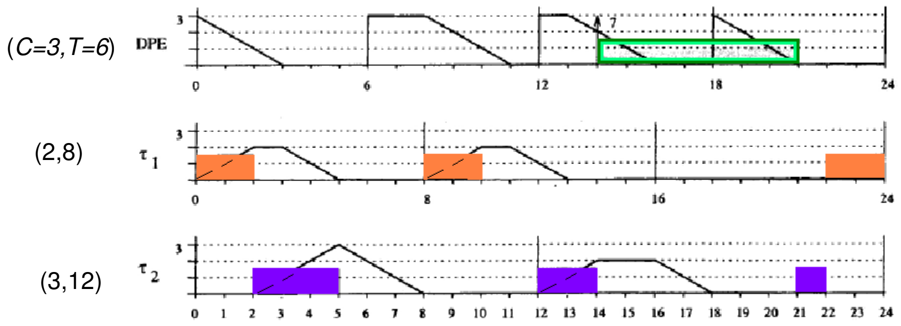
Dynamic Priority Exchange Server

- Whenever there are no aperiodic tasks to be served, the Dynamic Priority Exchange Server exchanges priorities with periodic tasks and wastes as little time as possible.
- It also facilitates using the spare time when periodic tasks do not use their worst-case times.

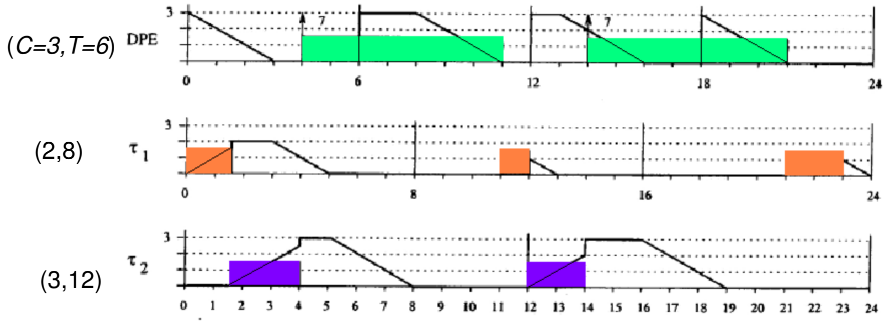
Dynamic Priority Exchange Server Scheme

- The DPE server has a specified period T_s and capacity C_s
- At the beginning of each period, the server's *aperiodic* capacity is set to C_s^d, where *d* is the deadline of the current server period.
- Each deadline *d* associated with the instances (completed or not) of the i^{th} periodic task has an aperiodic capacity C^d_{Si} initially set to 0.
- Aperiodic capacities (those greater than 0) receive priorities according to their deadlines and the EDF algorithm
  - Like all the periodic task instances
  - Ties are broken in favor of aperiodic requests
- Whenever the highest-priority entity in the system is an aperiodic capacity of *C* units of time
  - Any aperiodic requests are served until they complete or the capacity is exhausted
  - If there are no aperiodic requests pending, the periodic task having the earliest deadline is executed: a capacity equal to the execution is added to the aperiodic capacity of the task deadline and is subtracted from *C*
    - i.e. the deadlines of the highest-priority capacity and the periodic task are exchanged
  - If neither aperiodic requests no periodic tasks are pending, the processor becomes idle and the capacity *C* is consumed until it is exhausted

Dynamic Priority Exchange Server Example



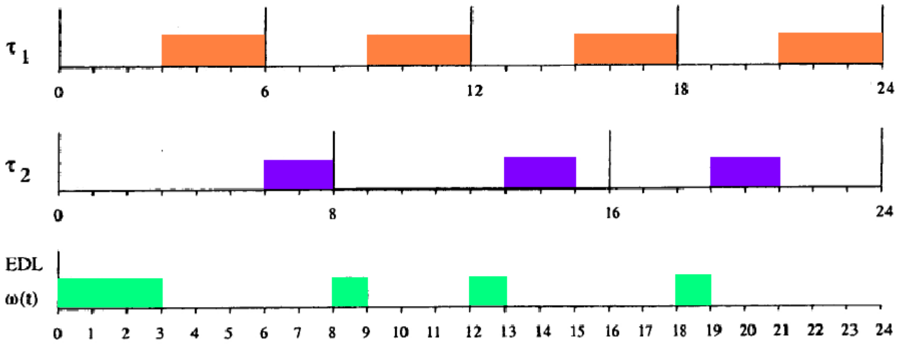
Reclaiming Spare Time



Earliest Deadline Late (EDL) Server

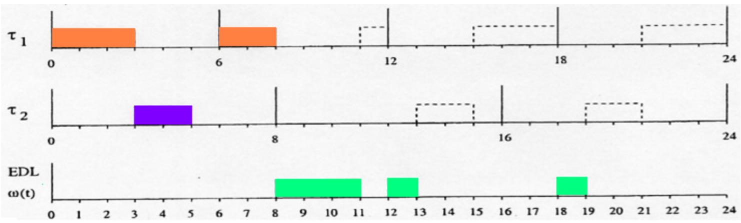
- The Total Bandwidth Server is conservative. There is **slack time** that could have been utilized to give the aperiodic tasks better response.
- EDL Server: Use the idle times of the EDL schedule to execute aperiodic requests as soon as possible.
  - When there are no aperiodic activities, periodic tasks are scheduled according to EDF.
  - When a new aperiodic request arrives and no prior aperiodic is still active, the idle times of an EDF scheduler applied to the current periodic taskset are computed and used to schedule the aperiodic requests pending.

Earliest Deadline Late Availability

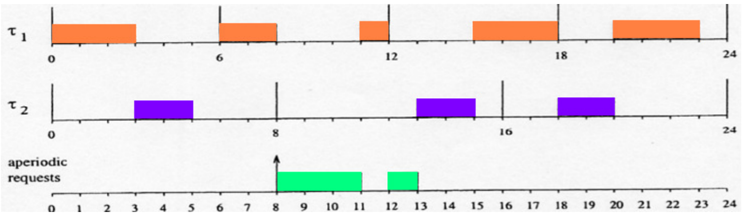


Availability Function Under EDL

Earliest Deadline Late Server Example



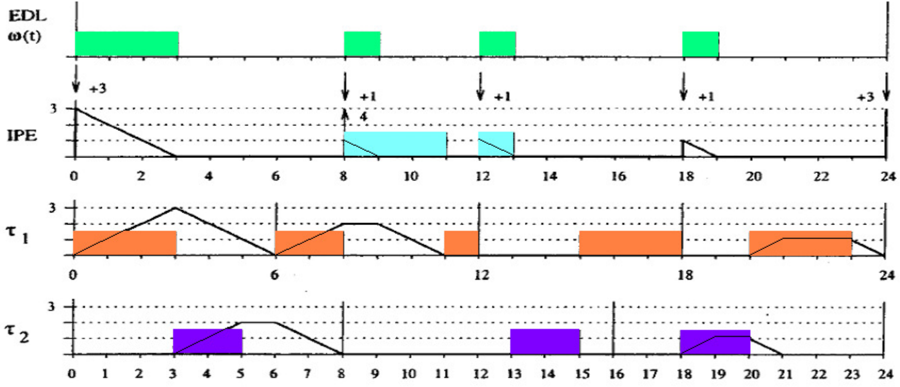
Idle times available at time *t* = 8 under EDL.



Schedule of the aperiodic request with the EDL server.

Improved Dynamic Priority Exchange Server

- EDL is overhead-intensive.
- In the Improved Dynamic Priority Exchange scheme, the Dynamic Exchange Server is modified to use the idle times of EDL, yielding:


EDL
ω(t)
IPE
τ 1
τ 2