The 5th EuroSys Doctoral Workshop

(EuroDW 2011)

Hierarchical Real-Time Systems for Imprecise Computation Model

Guy Martin Tchamgoue

Department of Informatics,

Gyeongsang National University, South Korea

Real-Time Systems Laboratory @ GNU

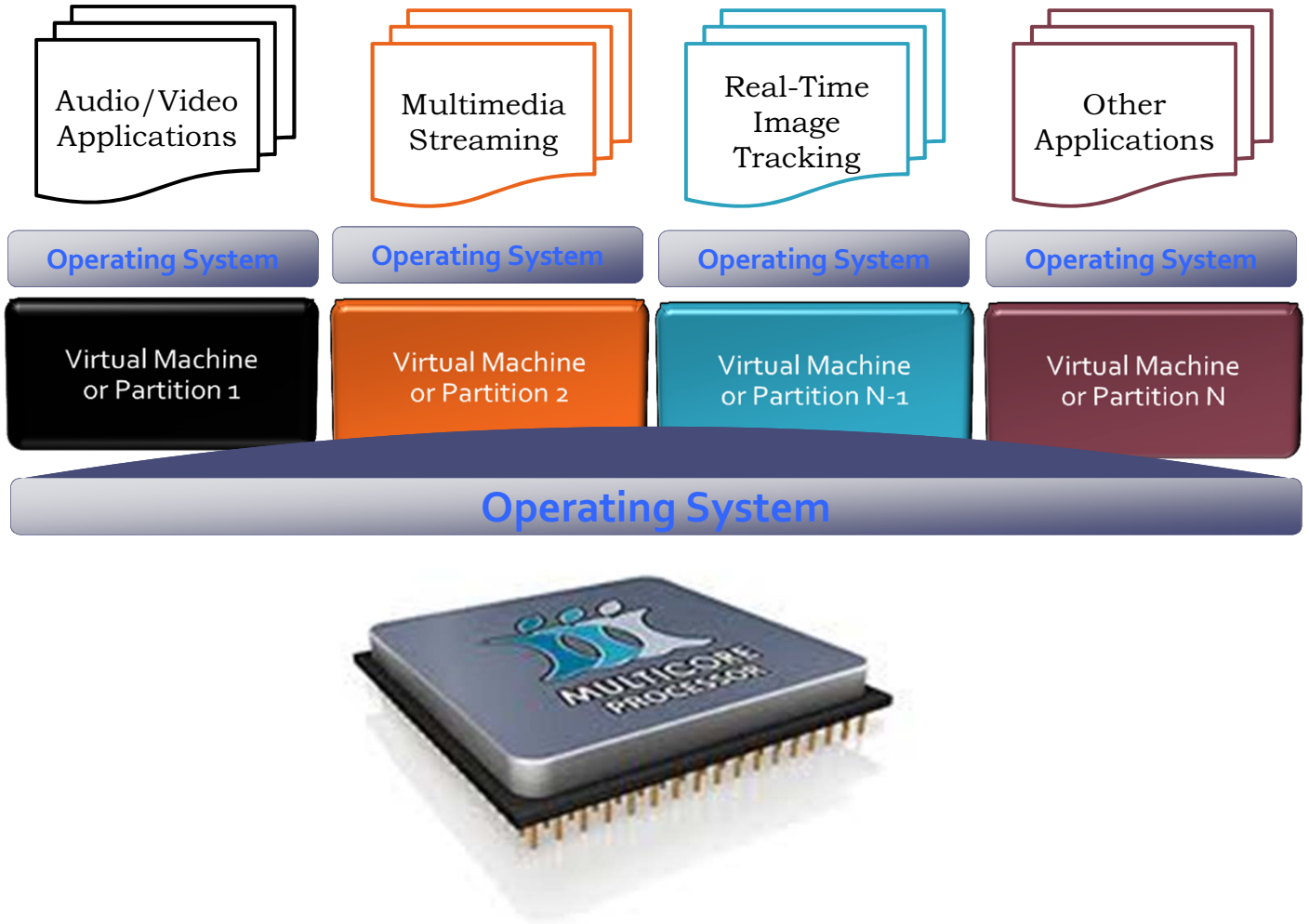
Agenda

- Problem
- An Application
- Proposed Solution
- Preliminary Results
- Ongoing Work

Problem

- Statement
  - To build large complex real-time systems from simpler ones
    - Flexibility and Reliability
      - To handle both hard and soft real-time systems
      - To enhance the resource utilization on system overload
      - To facilitate the development and deployment process

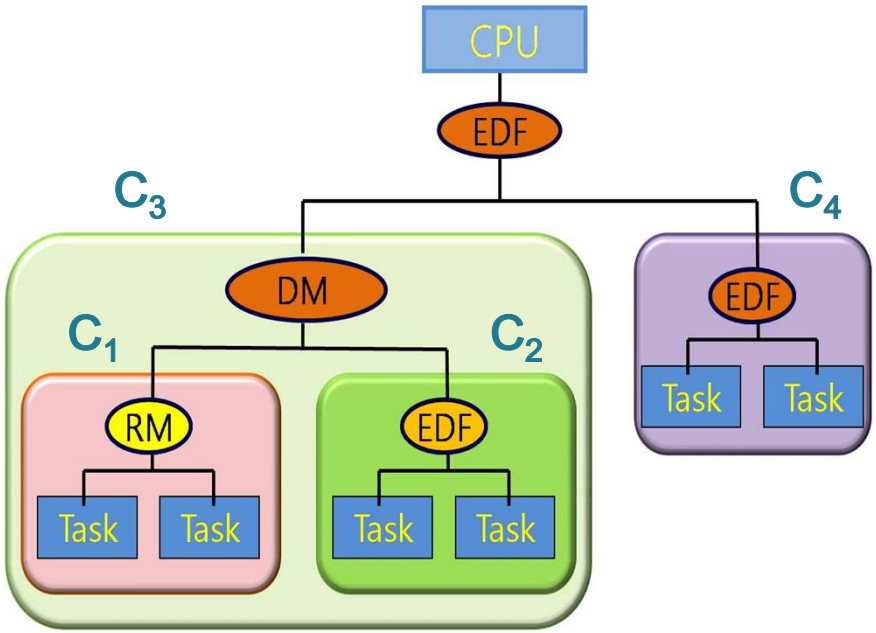
An Application



Compositional Scheduling Framework

- Hierarchical resource sharing among components under different scheduling policies

- Higher-level schedulers:
  - Consider each component as a scheduling unit
  - Do not need to know about the internal complexity of its components



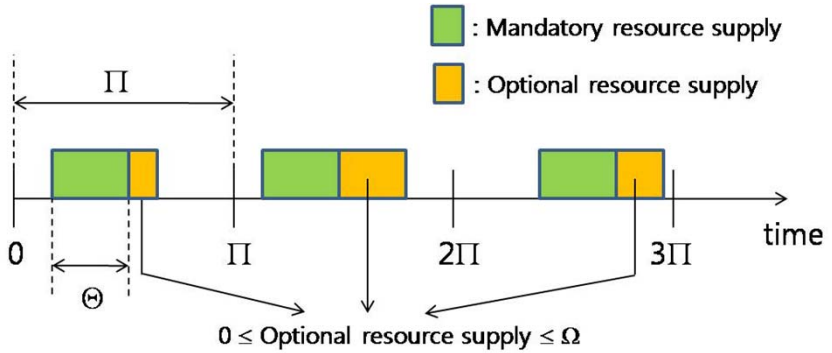
- Lower-level schedulers:
  - Focus on local scheduling for a given resource supply

Preliminary Results

- **Publications**
  - Tchamgoue et al., "*Compositional Scheduling Framework for Imprecise Computation Model*", Work-in-Progress Session, RTAS'10, April 2010
  - Tchamgoue et al., "*Hierarchical Real-Time Scheduling Framework for Imprecise Computations*", 8th IEEE/IFIP EUC, December 2010

- **System Model**
  - **Imprecise Computation** Tasks: $T = \{\tau_i (p_i, m_i, o_i, f_i)\}$
    - p_i : period
    - m_i : execution time of the mandatory part
    - o_i : execution time of the optional part
    - f_i : reward function for optional execution

- **The Imprecise Resource Model**: $\Gamma(\Pi, \Theta, \Omega)$
  - Π: resource period
  - Θ: resource allocation for mandatory part
  - Ω: resource allocation for optional part
  - $\Gamma(\Pi, \Theta, \Omega)$ periodically supplies at least Θ time units and at most $\Theta + \Omega$ time units.
  - Example:

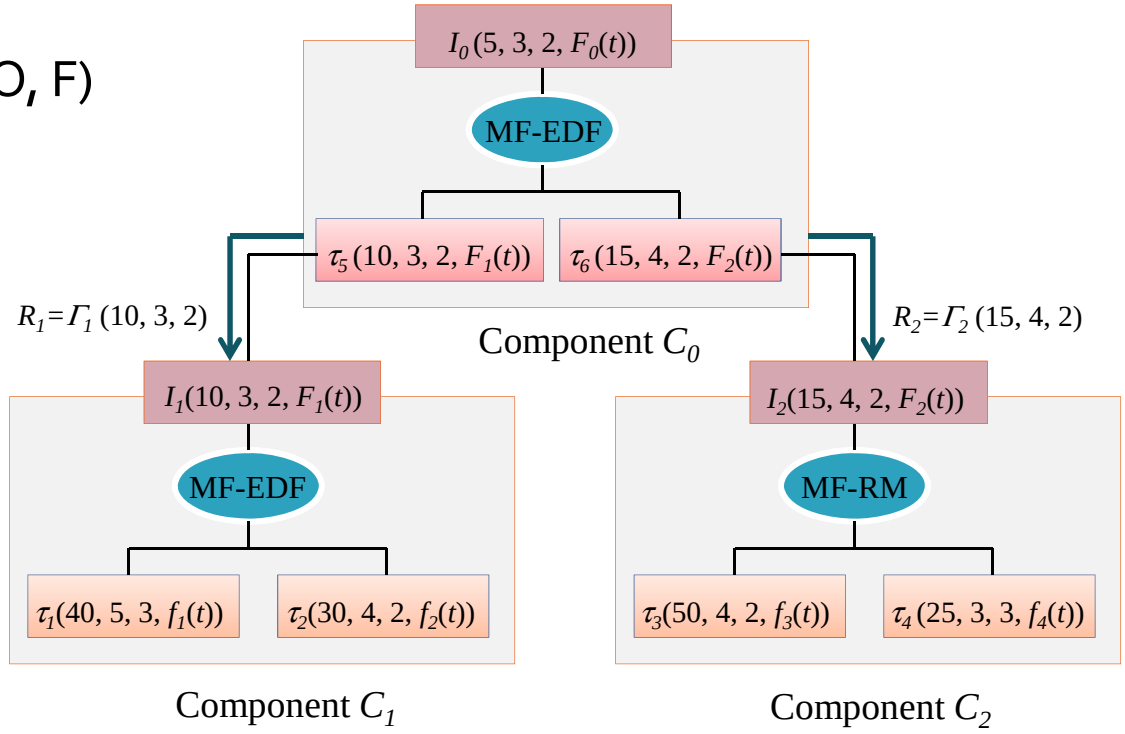


  - The resource supply of $\Gamma(\Pi, \Theta, \Omega)$

$$\Theta \le \text{supply}_{\Gamma}(k\Pi, (k+1)\Pi) \le \Theta + \Omega, \qquad \text{where } k = 0,1,2,\ldots$$

The Interface Model

- Specifying the real-time requirement of a component
- Interface Model I(P, M, O, F)
  - P : period
  - M : mandatory part requirement
  - O : optional part requirement
  - F : reward function of optional execution



- **Other Results**
  - Schedulability Analysis
    - To guarantee the minimum requirements of a component
  - Utilization Bound
    - To determine the largest possible utilization bound that makes a component schedulable
  - Scheduling Algorithm with Guarantee of Reward
    - To provide a minimum reward guarantee to an upper layer that provides resources to lower layers

Ongoing Work

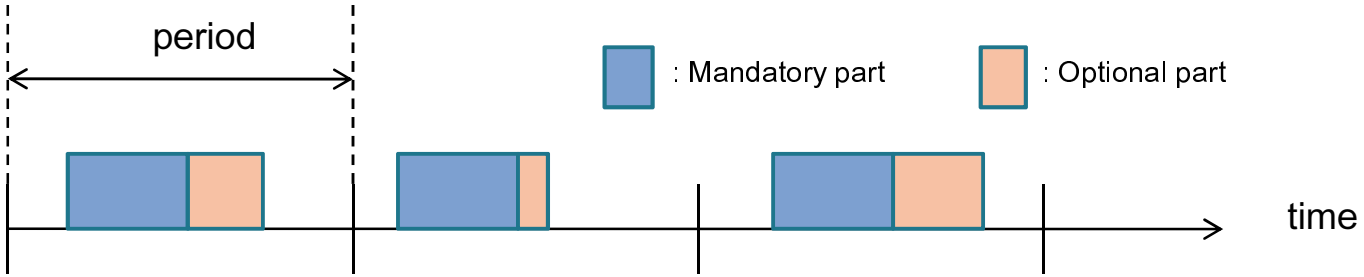
- **Component Interface Generation**
  - How to optimally compute the resource and interface parameters
- **Component Reward function**
  - To derive the reward function for the component's interface
- **Optimal scheduling of optional parts**
  - To schedule optional parts in order to maximize the total reward of a component

EuroDW 2011

Thank you for your attention.

Imprecise Computation* (IC) Model

- Proposed for flexible scheduling of hard deadline tasks
- An IC task consists of:
  - A mandatory subtask for an acceptable result, and
  - An optional subtask for
    - refining the mandatory task result ,and
    - improving the quality of the task according to the amount of execution time.



* J. W.-S. Liu, K.-J. Lin, W.-K. Shih, A. C.-S. Yu, C. Chung, J. Yao, and W. Zhao, "Algorithms for scheduling imprecise computations," *IEEE Computer*, vol. 24, no. 5, pp. 58–68, 1991.