Dynamic Placement for Clustered Web Applications

Written by A. Karve et al. at IBM Watson Research Center

Presented by Chen Zhang

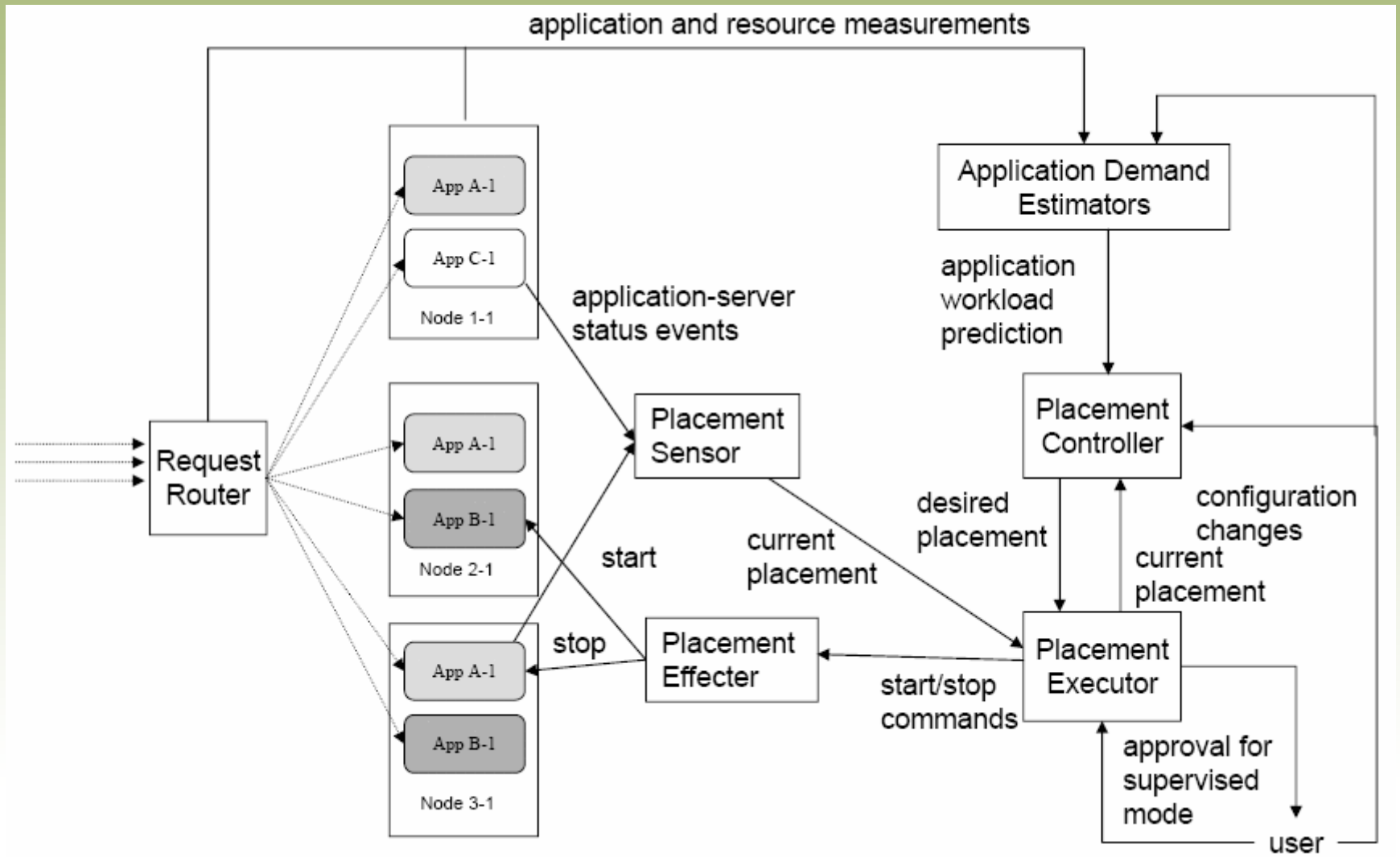
2006-11-22

Problem and Objectives

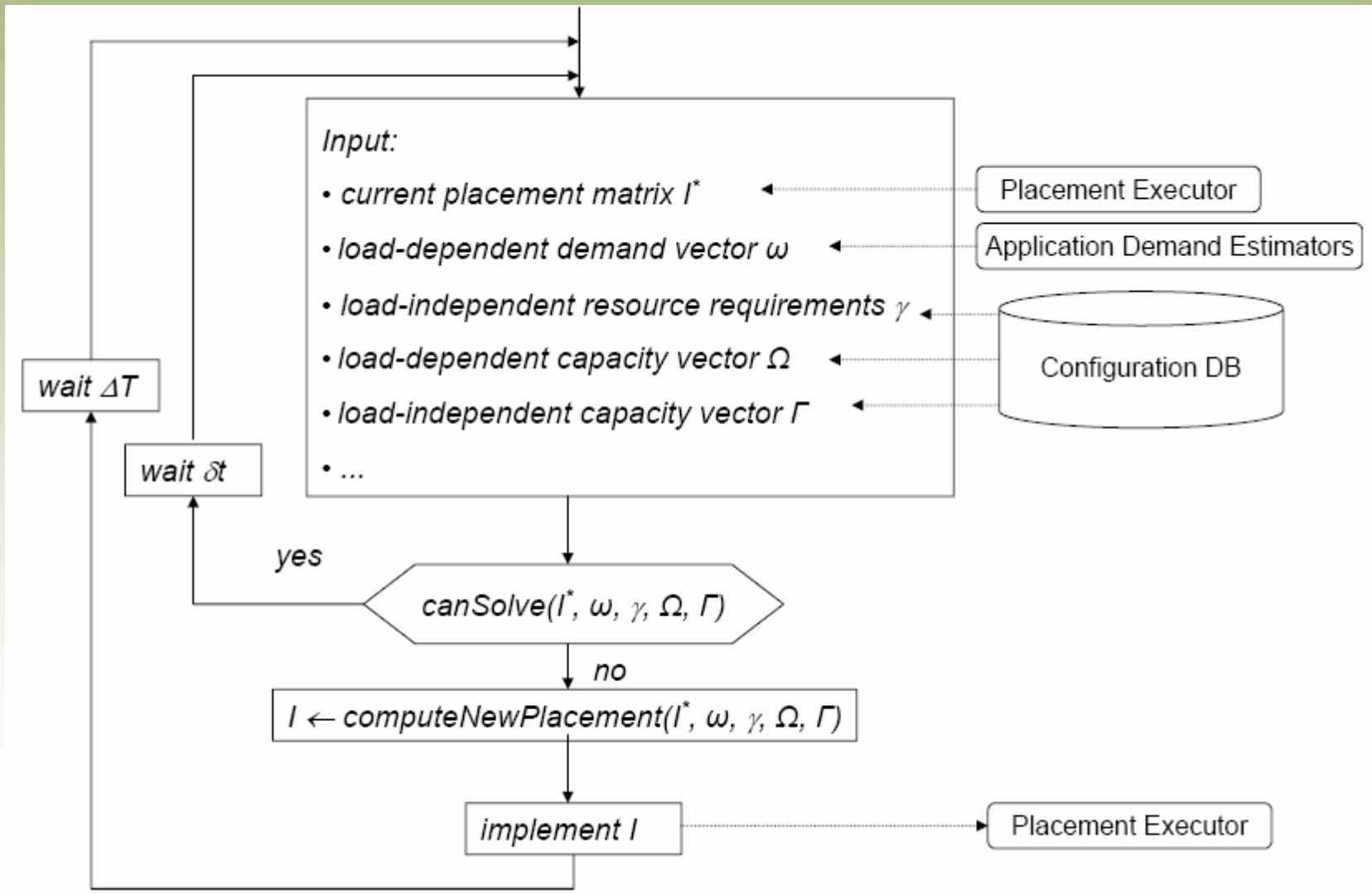
- Problem
  - Need to run multiple instances of applications
  - Limited per server capability
  - Server and application cluster heterogeneity
    - Different capability, different resource demand
    - Hardware or software requirement mismatch between server and some application cluster
- Objectives
  - Satisfy application cluster resource demand
  - Place application instances to balance server load

Define: Server capacity & resource demand
Load-independent: Memory
Load-dependent: CPU

Proposed Solution



Placement Controller



Optimization problem

Let L be a matrix of real valued *load placement* and $\rho = \frac{\sum_m \sum_n L_{m,n}}{\sum_n \Omega_n}$.

$$\begin{array}{ll} \text{(i)} & \max \sum_m \sum_n L_{m,n} \\ \text{(ii)} & \min \sum_m \sum_n |I^*_{m,n} - I_{m,n}| \\ \text{(iii)} & \min \sum_n (\frac{\sum_m L_{m,n}}{\Omega_n} - \rho)^2 \end{array} \quad (1)$$

s.t:

$$\forall_n \sum_m I_{m,n} \gamma_m \leq \Gamma_n \quad (2)$$

$$\forall_n \sum_m L_{m,n} \leq \Omega_n \quad (3)$$

$$\forall_m \sum_n L_{m,n} \leq \omega_m \quad (4)$$

$$\forall_m \forall_n I_{m,n} = 0 \Rightarrow L_{m,n} = 0 \quad (5)$$

$$\forall_m \forall_n \mathcal{R}_{m,n} = \text{false} \Rightarrow I_{m,n} = 0 \quad (6)$$

$$\forall_m \text{managed}_m = \text{false} \Rightarrow \forall_n I_{m,n} = I^*_{m,n} \quad (7)$$

3-part Heuristic Algorithm

- Residual placement
  - No previous placement
- Incremental placement
  - Based on previous placement
  - Do residual placement on unsatisfied demand
- Rebalancing placement
  - Shift load from overloaded to under-loaded node

Algorithm variants

- Basic Algorithm (BA)
- Load-reduction algorithm (LRA)
  - Not at peak capacity
- Multiple-runs algorithm (MRA)
  - Better balanced load
  - Previous placement affects new placement

Concerns

- Why not a “clever” admission control?
- Is it feedback control?
- When does it work? Hidden assumptions?
- What about the waiting time ΔT, σt ?
- Too J2EE specific, generalization?
  - capability and demand definition
  - Easy to deploy and migrate
  - Able to poll for multiple input
- Benefits for having LRA MRA
- Future work mentioned by author
  - Placement change
  - Prioritize applications
  - Equalize response time rather than CPU