Some Initial Performance Characteristics of Three Architectural Styles

Håkan Grahn and Jan Bosch

Dept. of Computer Science
University of Karlskrona/Ronneby
Sweden

Outline

- Software architecture
- Architectural style
- Characterization methodology
- Preliminary results
- Conclusions

Software Architecture

- Top level description/design of the system
- Identification of core abstractions
- Early in the design process
- Functional requirements
- Quality attributes
  - development Q.A.s, e.g., maintenance and reuse
  - operational Q.A.s, e.g., robustness, reliability, and performance
- Quality attributes are often constrained by the architecture

Architectural design is concerned with the *balancing* of quality attributes!

Architectural Style

- Defines a family of systems in terms of pattern of organization
- We have studied
  - pipes&filters
  - layered
  - blackboard
- Try to answer the following questions:
  - For each studied architectural style, what are the variables of the style that influence the performance and what are the performance characteristics?
  - How do the performance characteristics of the studied architectural styles relate to each other?

Characterization Methodology

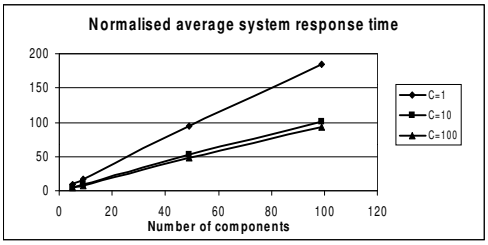
- Event-driven simulation
  - Set of components
  - Connection between the components
  - Events and event handlers
- Performance indicators
  - Throughput
  - System response time
  - Queue time for events
  - Queue length for components

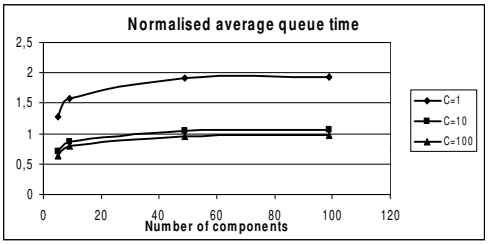
Our intention is to present *relative* performance characteristics

Architectural Parameters

- General , i.e., style independent parameters
  - Number of components
  - Computation-to-communication ratio
  - Blocking vs. non-blocking communication
- Style specific parameters
  - Pipes&filters: one-to-one, one-to-many, many-to-one (branching)
  - Layered: pure or non-pure style
  - Blackboard: number of components affected by an update (not done yet)

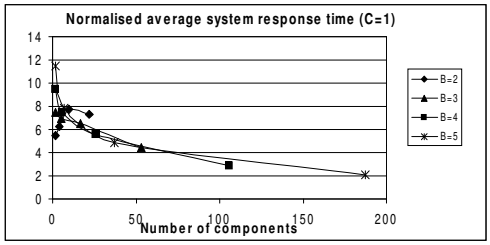
Pipes & Filters Style

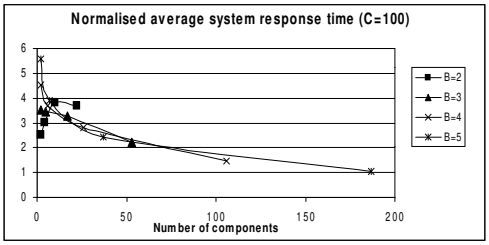




- The system response time increases to the square of the number of components
- The average queue time increases proportionally to the number of components

Pipes & Filters Style -- Branching System Response Time



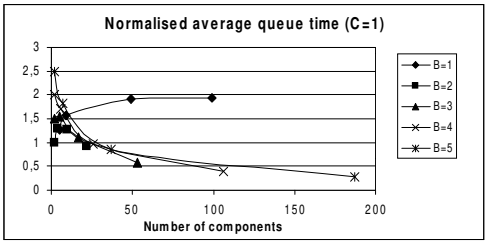


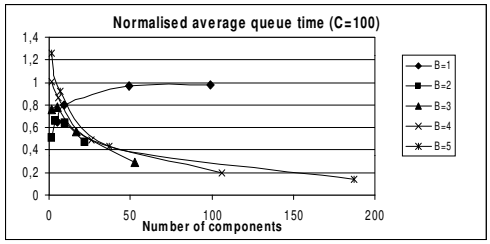
The normalised system response times

- increase for small branching factors (B=2 and B=3) when the number of components is small (<10)
- decrease for branching factors larger than 3 or more than 10 components

when the number of components increases

Pipes & Filters Style -- Branching Queue Time



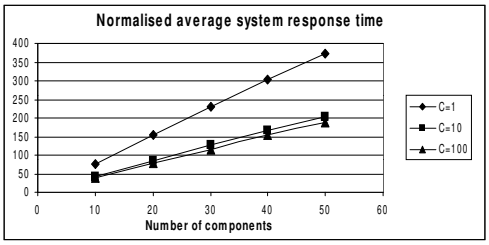


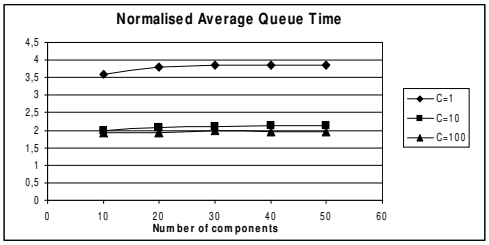
The normalised average queue time

- increases when B is small (2 or 3) and there is only a small number of components
- decreases when the number of components is larger than five

as the number of components increases.

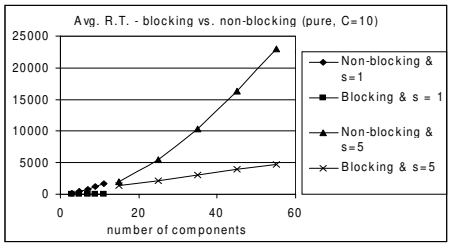
Layered Style

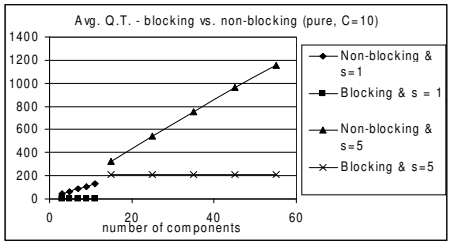




- The system response time increases to the square of the number of components
- The average queue time increases proportionally to the number of components

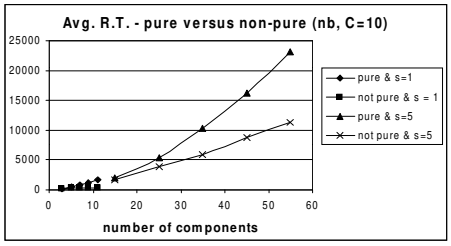
Layered Style -- Blocking vs. Non-blocking Communication

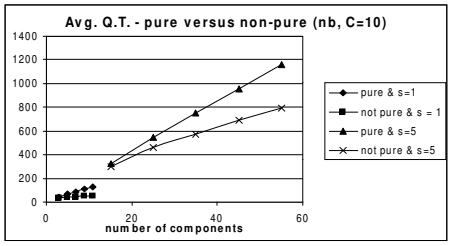




- The system response time increases to the square of the number of components for non-blocking communication, whereas it increases linearly for blocking communication
- The average queue time increases linearly when the number of components increases in the case of non-blocking communication, but it is constant for blocking communication

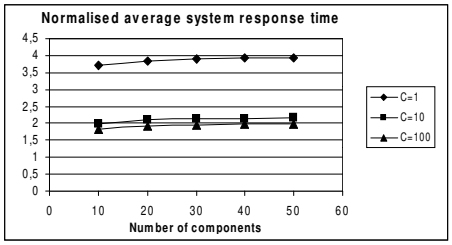
Layered Style -- Pure vs. Non-pure

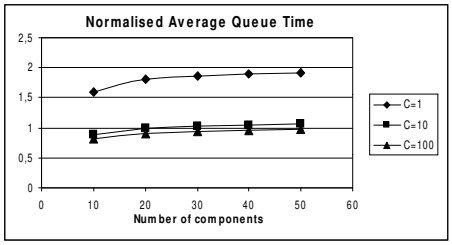




- The system response time is lower for the non-pure architecture and increases almost linearly with the number of components, whereas the response time develops relative to the square of the number of components for pure architectures
- The queue time increases proportionally to the number of components for both pure and non-pure architectures

Blackboard Style





- The average system response time increases proportionally to the number of components
- The average queue time increases proportionally to the number of components

Concluding Remarks

- Presented some initial performance characteristics of three architectural styles based on event-driven simulations
- System response time increases
  - to the square of the number of components for P&F and L
  - proportionally to the number components for BB
- The average queue time grows proportionally to the number of components (#components >= 30)
- Need validation against real applications