UNIVERSIDADE DO MINHO

Internal Conference on Computer Architecture
(ICCA’04)

Characterization of Workload suites for Performance Evaluation

Carlos Manuel Gomes Jardim

Overview

- ✓ The performance of a system is influenced by the characteristics of its hardware and software components as well as of the load it has to process.
- ✓ The performance of any type of system cannot be determined without knowing the workload, that is, the requests being processed.
- ✓ Workload characterization consists of a description of the workload by means of quantitative parameters and functions, in order to derive a model able to show, capture, and reproduce the behaviour of the workload and its most important features.

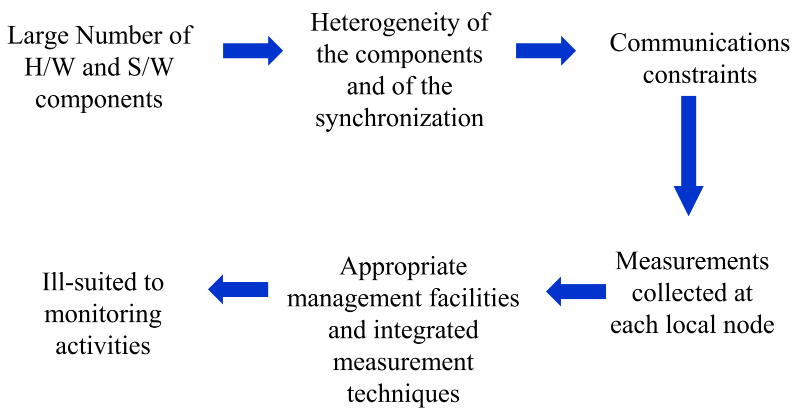
Overview (cont.)

- ✓ The approach commonly adopted for workload characterization is experimental, that is, based on the analysis of measurements collected on the system while the workload is being processed;
- ✓ Appropriate instrumentation has to be developed in order to ensure the quality of the measurements which have to adjust to the characteristics of the systems and of their workloads;
- ✓ the degree of intrusiveness and the overhead introduced by the instrumentation system have to be as low as possible in order not to perturb the behaviour of the system and of its workload

Distributed Systems

- ✓Collection of heterogeneous computers and processors connected via network;
- ✓Distribute computation among multiple host that can be in different locations;
- ✓Works together to accomplish a common goal;
- ✓Each processor has its own local memory;
- ✓Processor communicate with one another through various communications lines;

Workload Characterization



Large Number of H/W and S/W components

Heterogeneity of the components and of the synchronization

Communications constraints

Measurements collected at each local node

Appropriate management facilities and integrated measurement techniques

Ill-suited to monitoring activities

Client / Server environment (cont.)

✓Studies related to capacity planning, Load Balancing, resource allocation and scheduling, and congestion control policies, rely on an accurate characterization of the workload of client/server environments;

✓This characterization is also the basis for the definition of the input parameters of models, aimed, for instance, at performance prediction.

Workload Characterization (cont.)

Client / Server Environment

A client/server environment is typically composed of clients connected to servers through a network.

- ✓ Distributed file systems;
- ✓ Distributed databases;
- ✓ World Wide Web;
- ✓ Distributed multimedia systems.

Client / Server environment

Hierarchically structured into three layers:

- ✓ Client;
- ✓ Network;
- ✓ Server.

Depending on the considered layer, **workload** consists of the requests, as seen at the client or server layers, or of the packets, owing on the network.

Network Level

Software monitors, (e.g. *tcpdump)*, capture:

packets flowing on the network, and provide user level control of measurement collection;

- ✓ Filtering on a per host;
- ✓ Protocol;
- ✓ port basis;

Must be capture:

- ✓ Packet arrival times;
- ✓ Packet lengths;
- ✓ Source and destination host;
- ✓ Port number;

Network Level (cont.)

Hardware monitors. (e.g. *sniffers)*, capture the packets, owing on the network, and perform some preliminary analyses on them.

Must be gathered:

- ✓ Timing of the packet arrivals;
- ✓ Packet lengths,

together with events:

- ✓ Packet retransmissions

Client / Server Layer

Use of accounting routines provides measurements about processed requests. Events of interest could be capture modifying the source codes of client or server applications, by adding custom instrumentation.

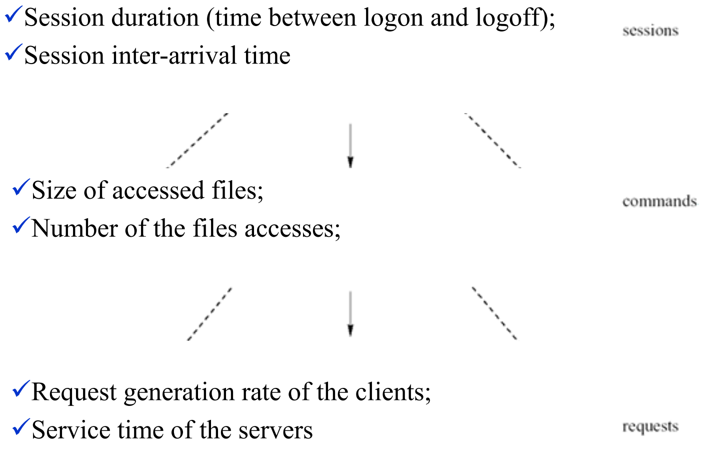
Information captured - Client:

- ✓ Arrival time of each request;
- ✓ Size of requested file;
- ✓ URL;
- ✓ Session;
- ✓ User;
- ✓ Client identifier.

Server:

- ✓ CPU and disk demands of each request;
- ✓ Access/modification time.

Networkload models



- ✓ Session duration (time between logon and logoff);
- ✓ Session inter-arrival time

sessions

- ✓ Size of accessed files;
- ✓ Number of the files accesses;

commands

- ✓ Request generation rate of the clients;
- ✓ Service time of the servers

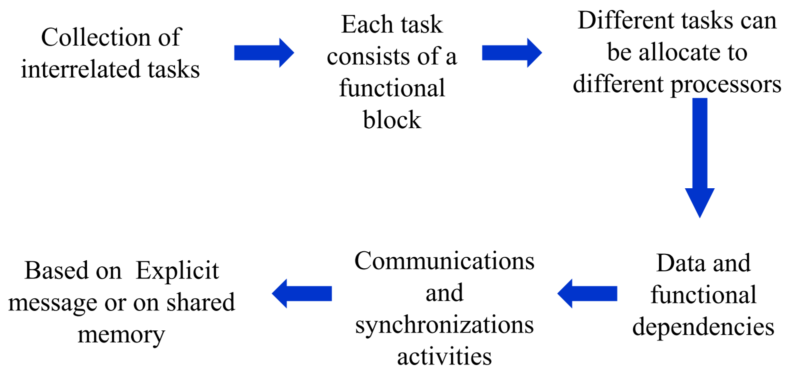
requests

Parallel Systems

Consists of multiple processes (or tasks) interacting together, allocated to different processors, and characterized by dependencies, synchronization constraints and precedence relationships arising from the distribution of the data and of the execution among distinct processors;

- ✓ Multi-computers: each processor has it own local memory;
- ✓ Multi-processor: processor share memory and clock;

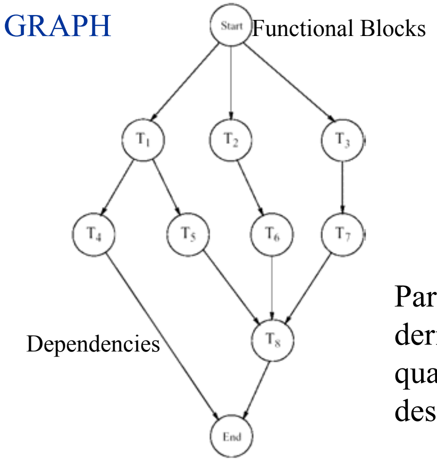
Workload Characterization



Collection of interrelated tasks → Each task consists of a functional block → Different tasks can be allocate to different processors → Data and functional dependencies → Communications and synchronizations activities → Based on Explicit message or on shared memory

Workload Characterization (cont.)

GRAPH



Parameters and metrics are derived in order to obtain qualitative and quantitative descriptions of the behaviour

Metrics

General classification of metrics can be done in terms of their dependence or independence from system architectures:

Static indices

- ✓Preliminary information about the behaviour of whole application;
- ✓Describe the complexity of the structure of an algorithm;

Dynamic indices

- ✓Describe the behaviour of an algorithm when it is executed on a given system;
- ✓Reflect how efficiently the parallelism is exploited;
- ✓Appropriate for evaluation of the match between algorithms and architectures;

Static indices

N

- ✓ Total number of nodes;

In-degree

- ✓ Avg. Number of direct predecessor of all the nodes;

Out-degree

- ✓ Avg. Number of direct successor of all the nodes;

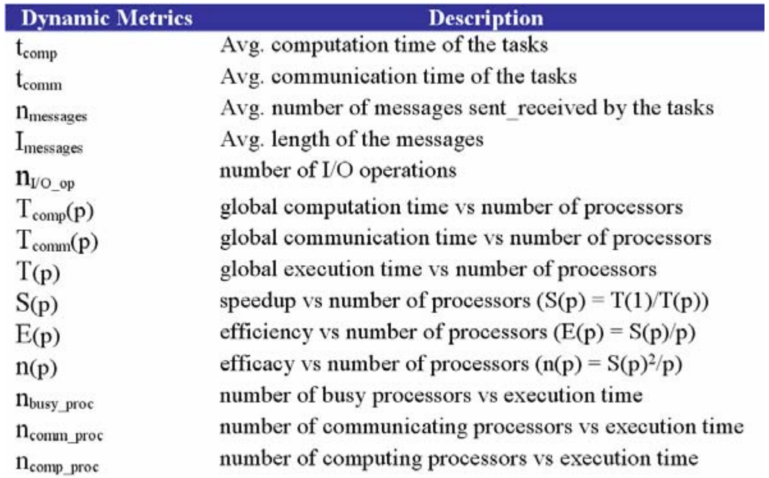
Depth

- ✓ Longest path between input and output nodes;

Maximum cut

- ✓ Max number of arcs taken over all possible cuts

Dynamic indices



Dynamic Metrics	Description
t_{comp}	Avg. computation time of the tasks
t_{comm}	Avg. communication time of the tasks
$n_{messages}$	Avg. number of messages sent_received by the tasks
$I_{messages}$	Avg. length of the messages
n_{I/O_op}	number of I/O operations
$T_{comp}(p)$	global computation time vs number of processors
$T_{comm}(p)$	global communication time vs number of processors
$T(p)$	global execution time vs number of processors
$S(p)$	speedup vs number of processors ($S(p) = T(1)/T(p)$)
$E(p)$	efficiency vs number of processors ($E(p) = S(p)/p$)
$n(p)$	efficacy vs number of processors ($n(p) = S(p)^2/p$)
n_{busy_proc}	number of busy processors vs execution time
n_{comm_proc}	number of communicating processors vs execution time
n_{comp_proc}	number of computing processors vs execution time

Conclusion

✓The methodologies and the techniques applied for constructing workload models are strictly related to the objectives of the studies as well as to the type of system to be tested;

✓The increasing number of hardware and software components interacting together makes the performance of such systems heavily dependent on the characteristics of the load;

✓it is necessary to identify a set of parameters able to capture and reproduce the behaviour and the evolution in time of the workload components processed by such systems

Conclusion

There are few steps that can be seen as a common basis for any workload characterization study:

- ✓ choice of the set of parameters able to describe the behaviour of the workload;
- ✓ choice of the suitable instrumentation that is use of existing performance monitoring tools;
- ✓ analysis of workload data;
- ✓ construction of static / dynamic workload models