SYSTEM LEVEL MODELLING AND PERFORMANCE ESTIMATION

EXPLORATION OF A DIGITAL AUDIO PLATFORM

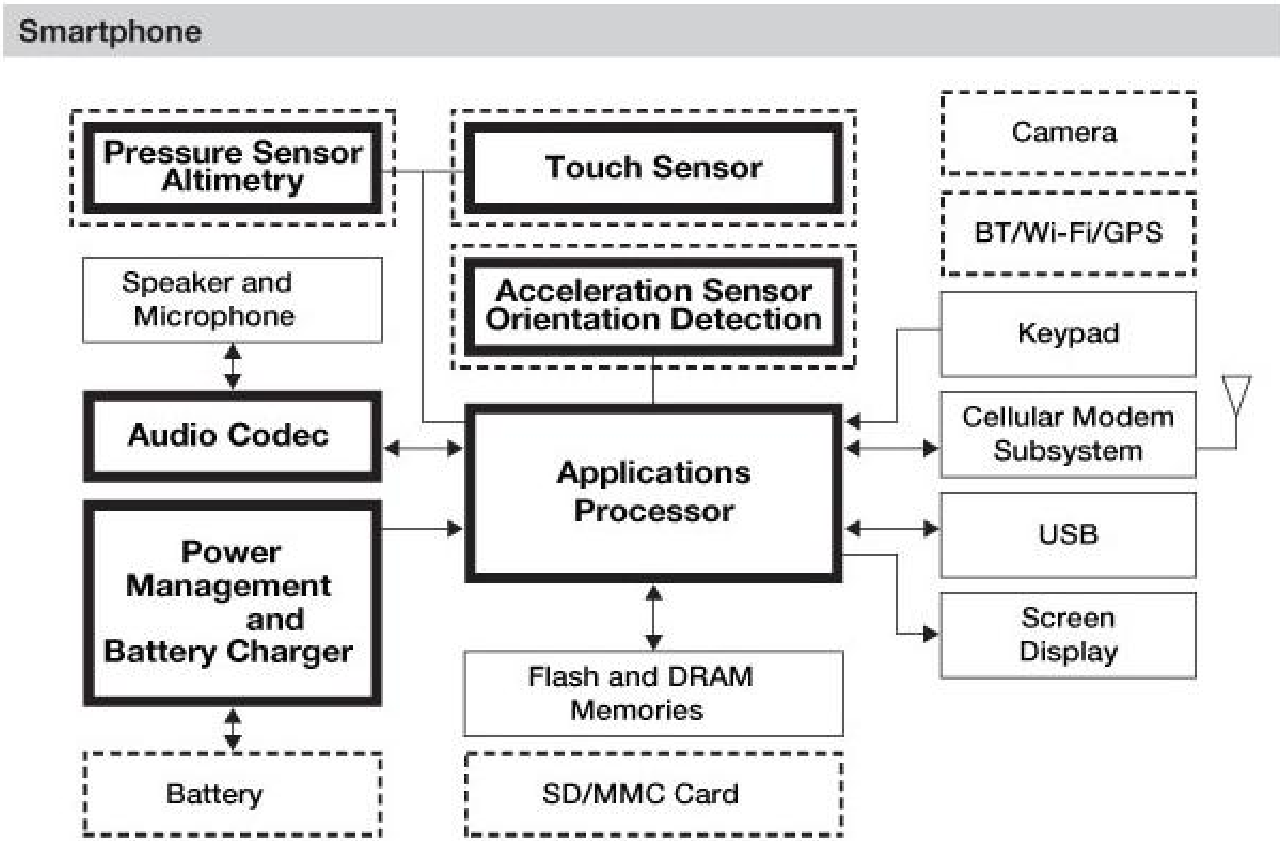
ANDERS SEJER TRANBERG-HANSEN AND JAN MADSEN

DTU Informatics
Department of Informatics and Mathematical Modeling

OUTLINE

MOTIVATION


SAMSUNG


Smartphone
Pressure Sensor
Altimetry
Touch Sensor
Camera
BT/Wi-Fi/GPS
Speaker and
Microphone
Acceleration Sensor
Orientation Detection
Keypad
Audio Codec
Applications
Processor
Cellular Modem
Subsystem
USB
Power
Management
and
Battery Charger
Screen
Display
Flash and DRAM
Memories
Battery
SD/MMC Card

MOTIVATION

- Increased design **complexity** is leading to sub-optimal implementations due to **time-to-market** constraints of systems.
- Difficult to get **feedback** to the consequences of a design choice before the system is realized – implying that in the early design phases **experience** of designers are a key element.
- Efficiency goals dominate leading to systems with less flexibility due to **insufficient exploration** of the design space.

PRESENTATION OUTLINE

- Overview of the Framework.
- Service Models.
  - Architecture Modelling
- Case-study: A Mobile Audio Processing Platform.
- Conclusion and Future Work.

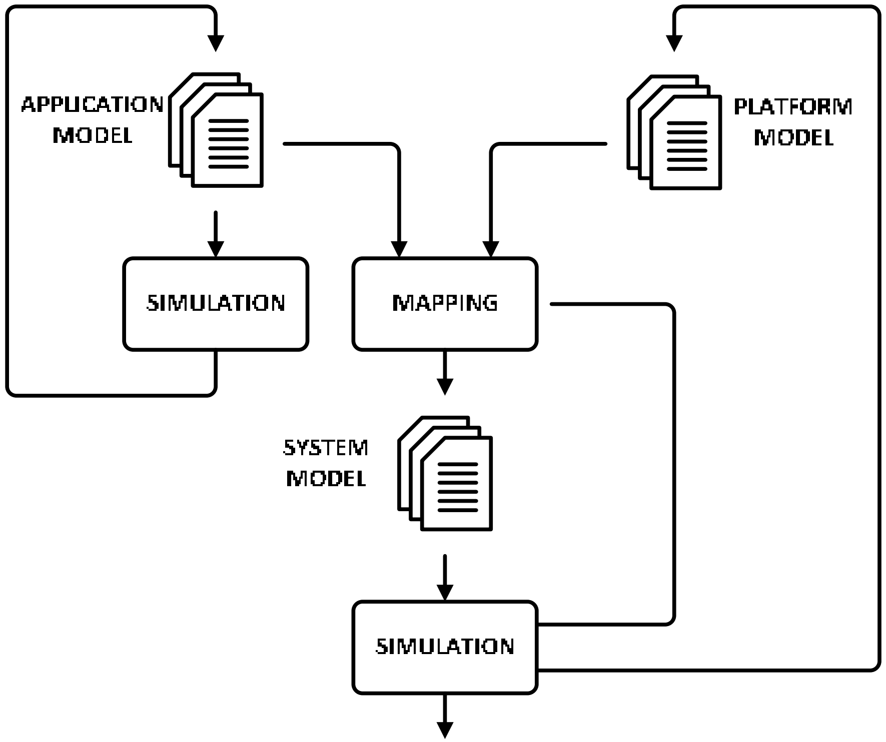
THE FRAMEWORK

A COMPOSITIONAL MODELLING FRAMEWORK FOR PERFORMANCE ESTIMATION

OBJECTIVES

- A framework for quantitative performance estimation for use in all design phases.
- The framework is a tool which can be used for exploration of the design space.
- Must be able to capture both applications and the platform onto which the application is executed.

OVERVIEW

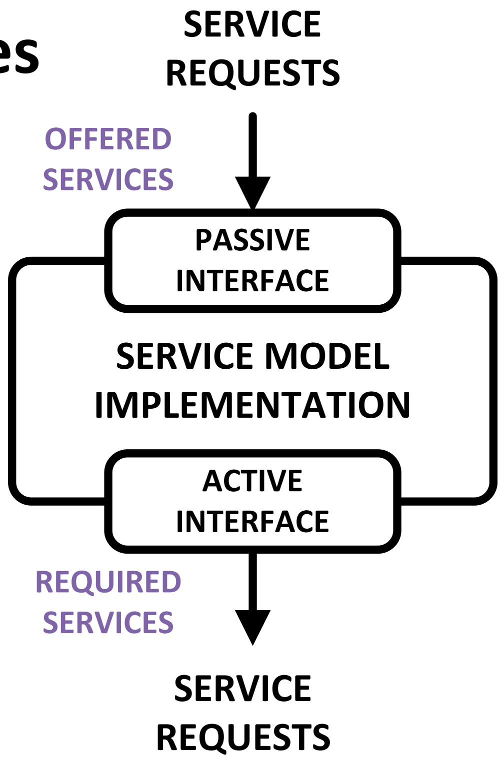

APPLICATION MODEL
PLATFORM MODEL
SIMULATION
MAPPING
SYSTEM MODEL
SIMULATION
TO IMPLEMENTATION FLOW

BASICS

SERVICE MODELS

SERVICE MODEL

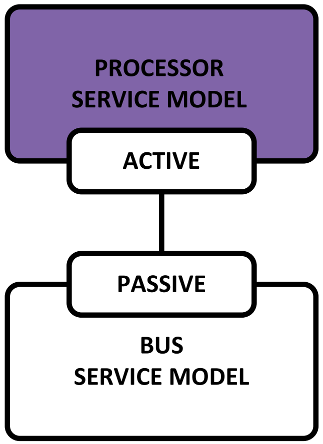
- The behaviour of a component is modelled by the availability of a set of services.
- A Service Model is composed of:
    1. One or more **Service Model Interfaces**
    2. One **Service Model Implementation**

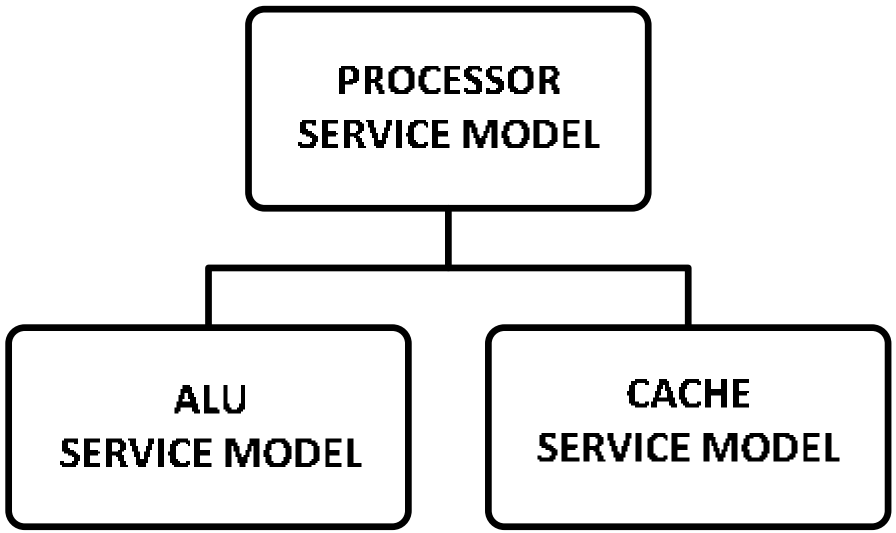


SERVICE MODEL

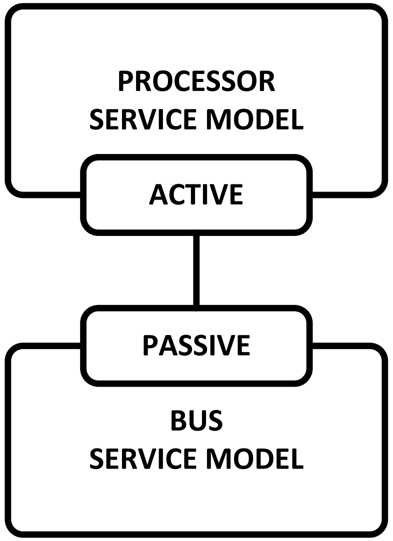
- A Service Model defines:
  1. The services offered.
  2. The implementation of the behaviour of the services.
  3. The resource requirements of the services.
  4. The inter-connection possibilities of the model.
  5. The latency of the services.

SERVICE MODEL COMPOSITION

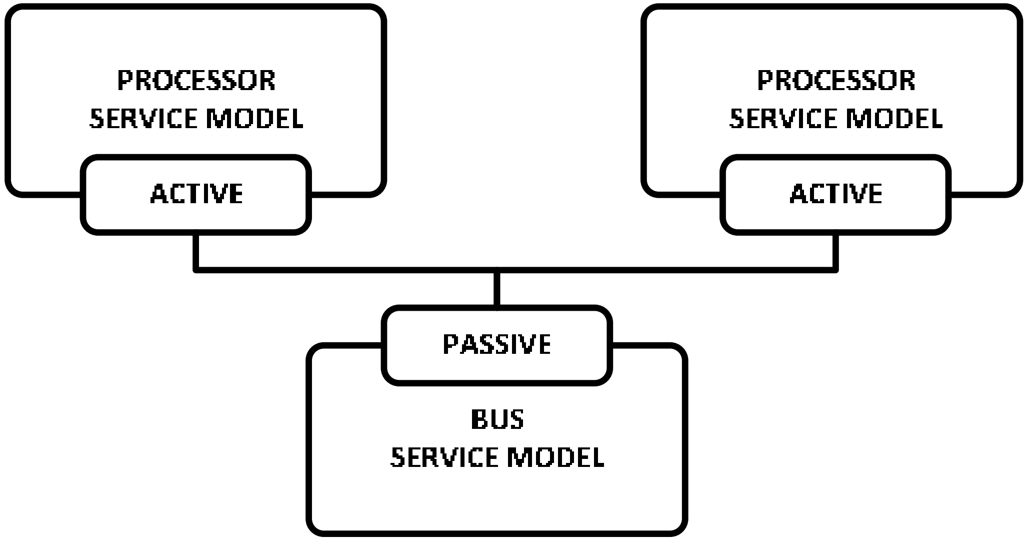




SERVICE MODEL COMPOSITION



SERIAL CONNECTION

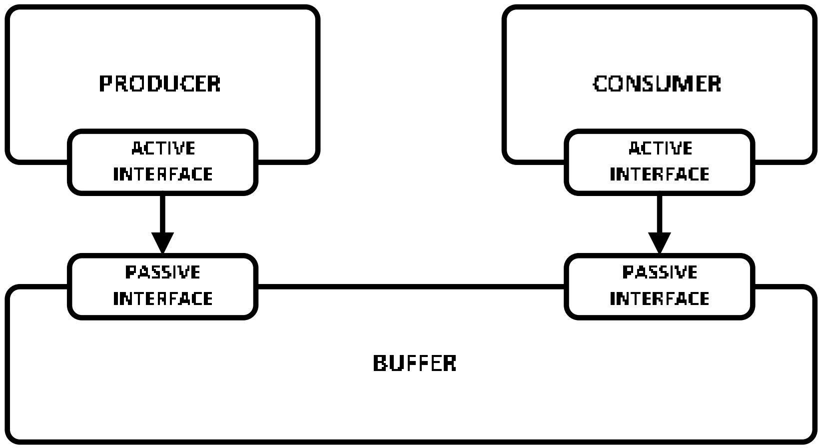


PARALLEL CONNECTION

THE FRAMEWORK

APPLICATION MODEL

APPLICATION MODEL



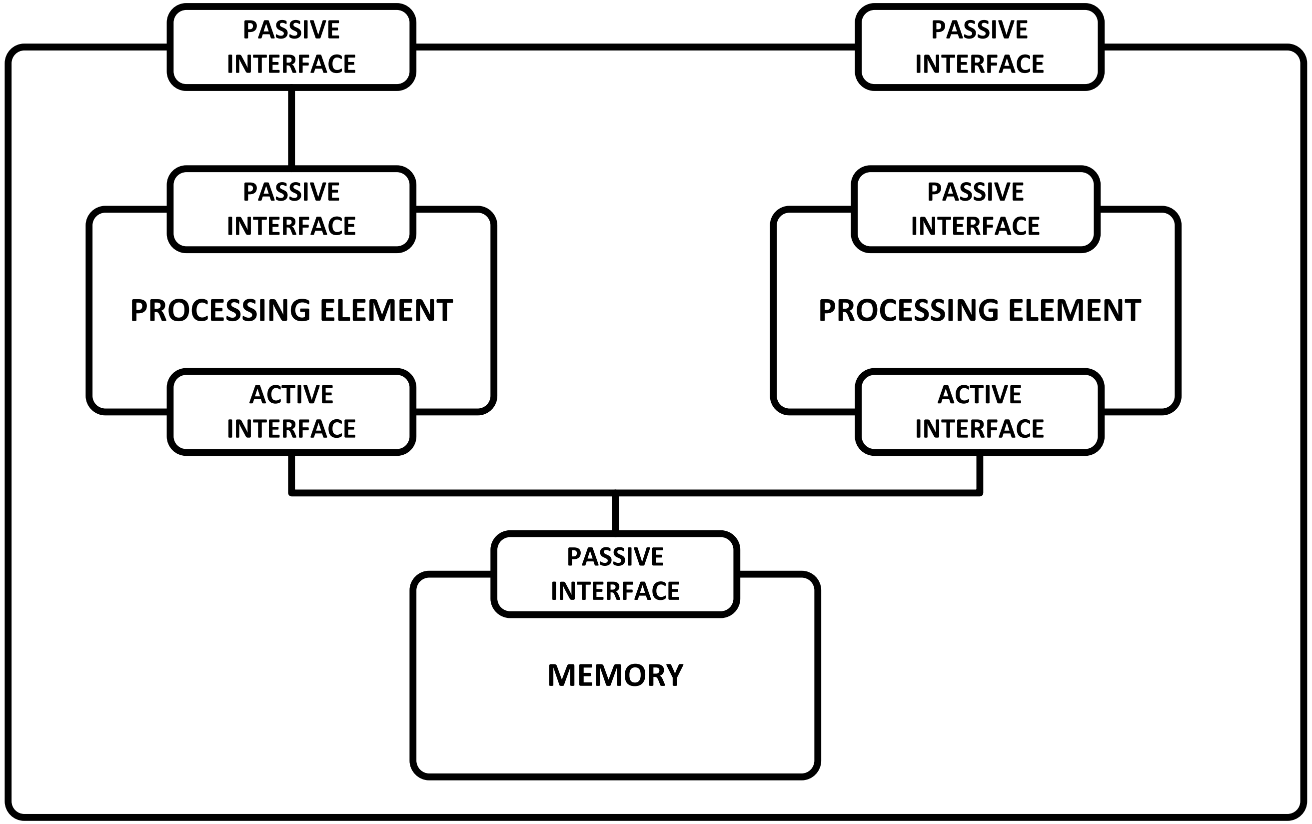

```
PRODUCER{
  ACTIVE_INTERFACE write;
  RUN(){
    for(int i=0; i < 63; i++){
      IntVal res;
      // CALCULATE RESULT
      ...
      write.request("WT", res);
    }
  }
}
```

```
CONSUMER{
  ACTIVE_INTERFACE read;
  RUN(){
    for(int i=0; i < 63; i++){
      IntVal res :=
        read.request("RD");
      // PROCESS DATA
      ...
    }
  }
}
```

THE FRAMEWORK

PLATFORM MODEL

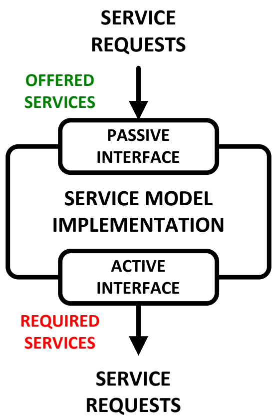
PLATFORM MODEL



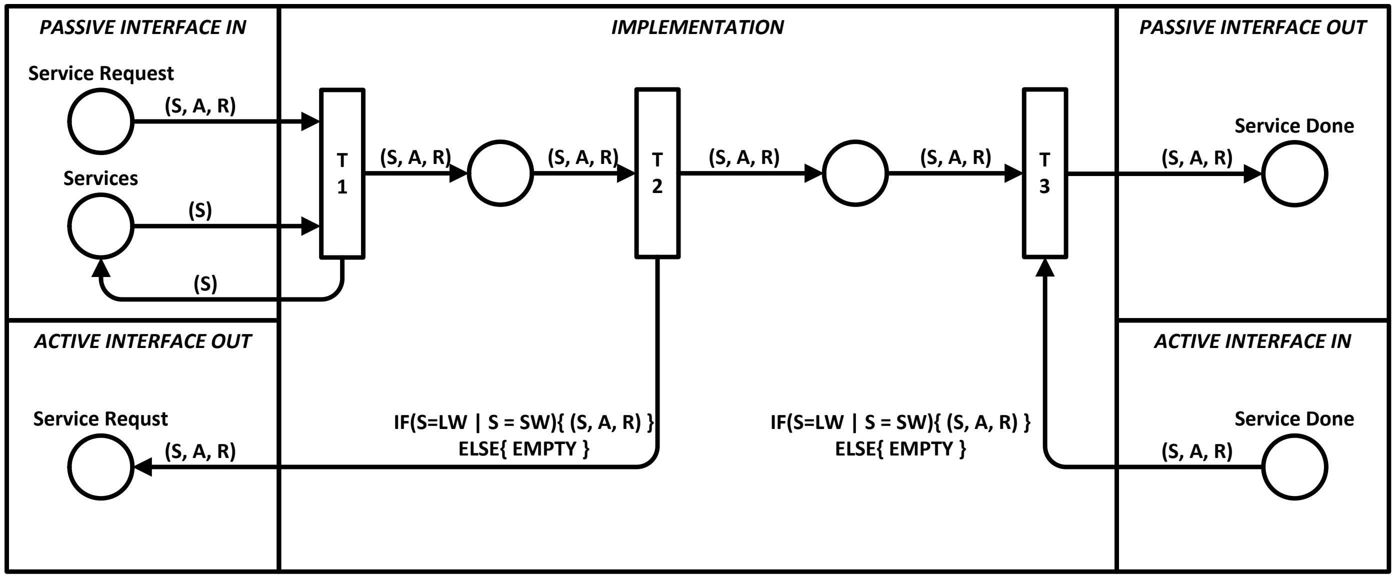
MODEL-OF-COMPUTATION

- Hierarchical Coloured Petri Nets.
  - Hierarchical.
  - Parallel activities.
  - Resource requirements and access to shared resources.
  - Data flow.
- Modified Execution Semantics.
- Quasi-static scheduling
- Two types of tokens only.

HCPN BASED SERVICE MODEL



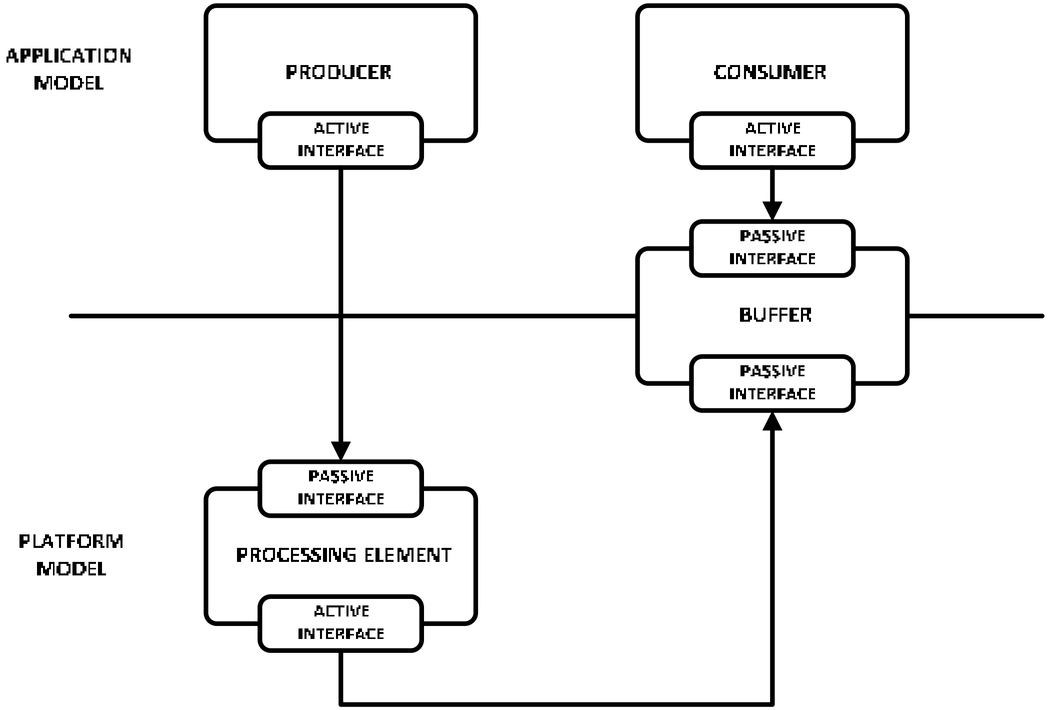
S = Service
A = ArgumentList
R = RequestNumber



THE FRAMEWORK

SYSTEM MODEL

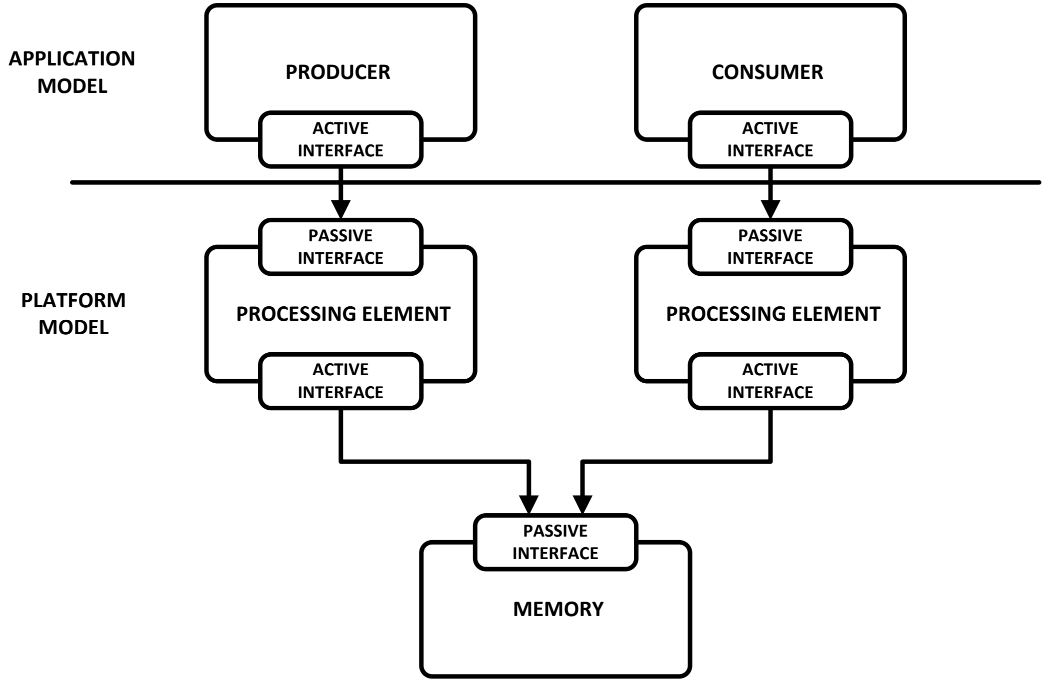
SYSTEM MODEL




```
PRODUCER{
  ACTIVE_INTERFACE sw;
  RUN(){
    for(int i=0; i < 63; i++){
      IntVal res;
      // CALCULATE RESULT
      sw.request(...);
      ...
      sw.request("WT", 0xF, res);
    }
  }
}
```

```
PRODUCER{
  ACTIVE_INTERFACE read;
  RUN(){
    for(int i=0; i < 63; i++){
      IntVal res :=
        read.request("RD", res);
      // PROCESS DATA
      ...
    }
  }
}
```

SYSTEM MODEL



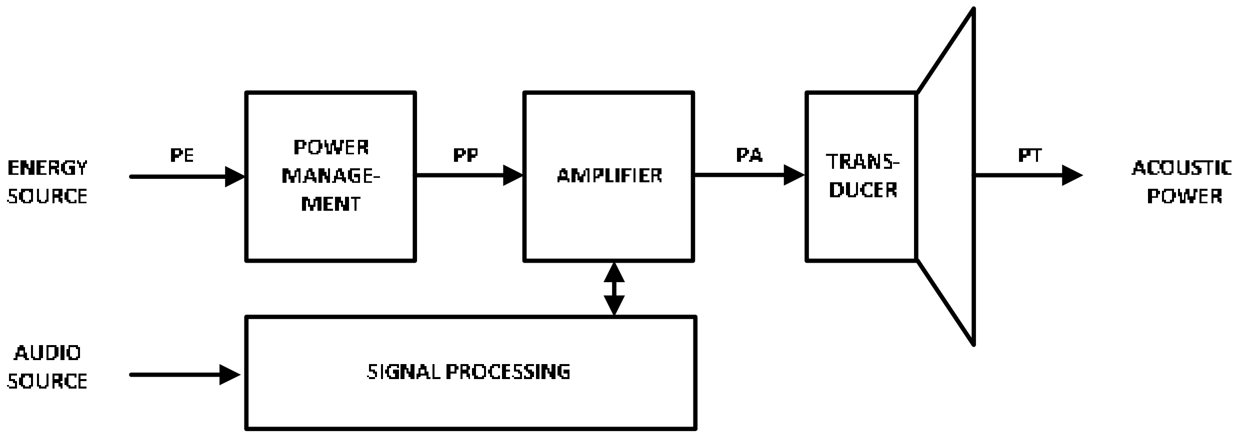

```
PRODUCER{
  ACTIVE_INTERFACE sw;
  RUN(){
    for(int i=0; i < 63; i++){
      IntVal res;
      // CALCULATE RESULT
      sw.request(...);
      ...
      sw.request("WT", res);
    }
  }
}
```

```
PRODUCER{
  ACTIVE_INTERFACE sw;
  RUN(){
    for(int i=0; i < 63; i++){
      IntVal res :=
        sw.request("RD", 0xA);
      // PROCESS DATA
      sw.request(...);
      ...
    }
  }
}
```

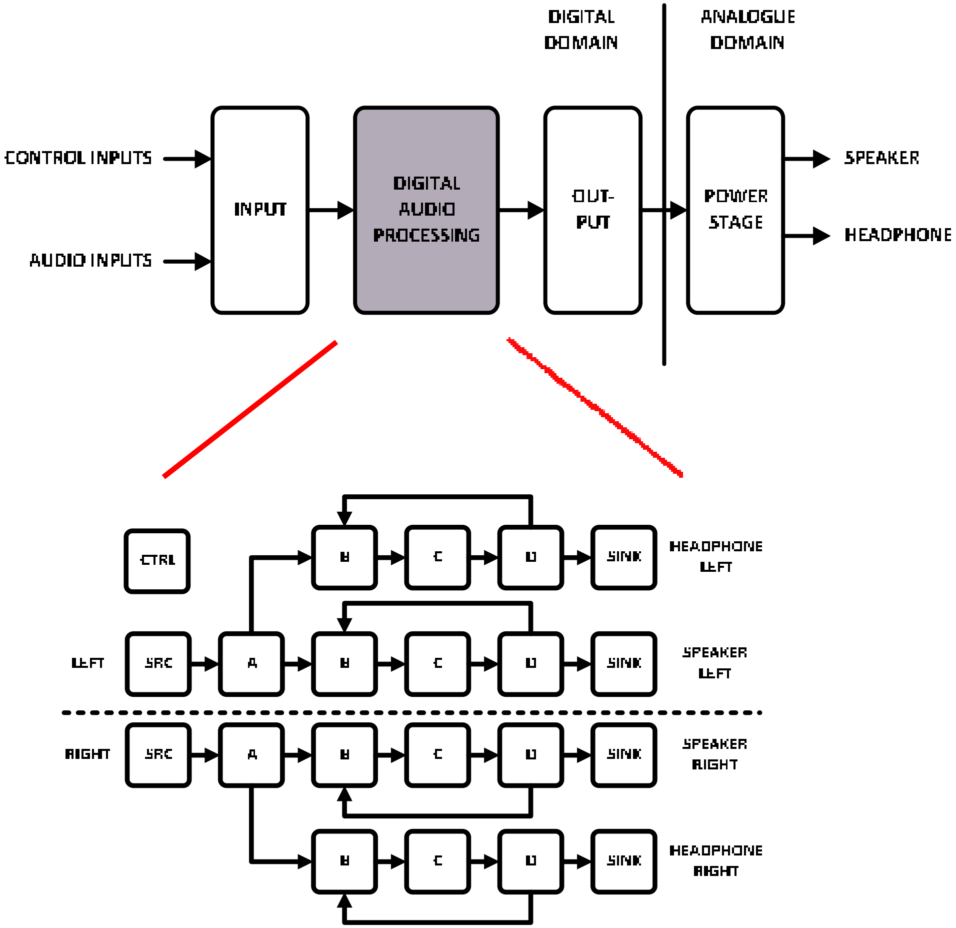
CASE-STUDY

A MOBILE AUDIO PROCESSING PLATFORM

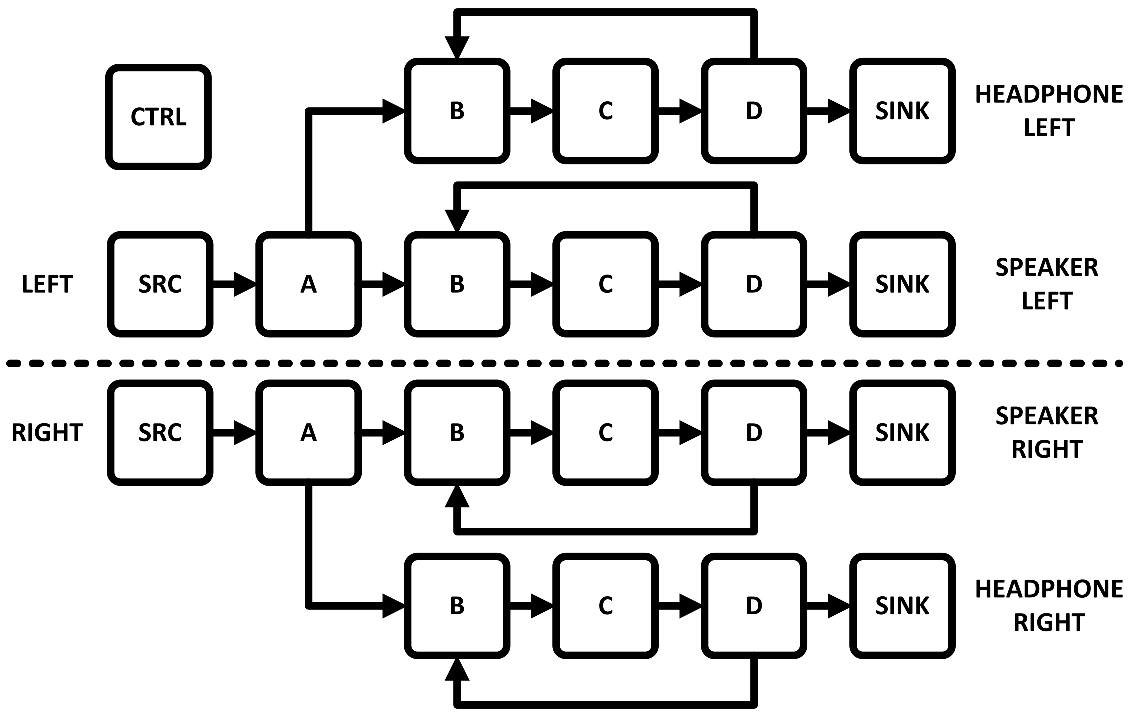
Bang & Olufsen ICEpower



MOBILE AUDIO PROCESSING



APPLICATION MODEL

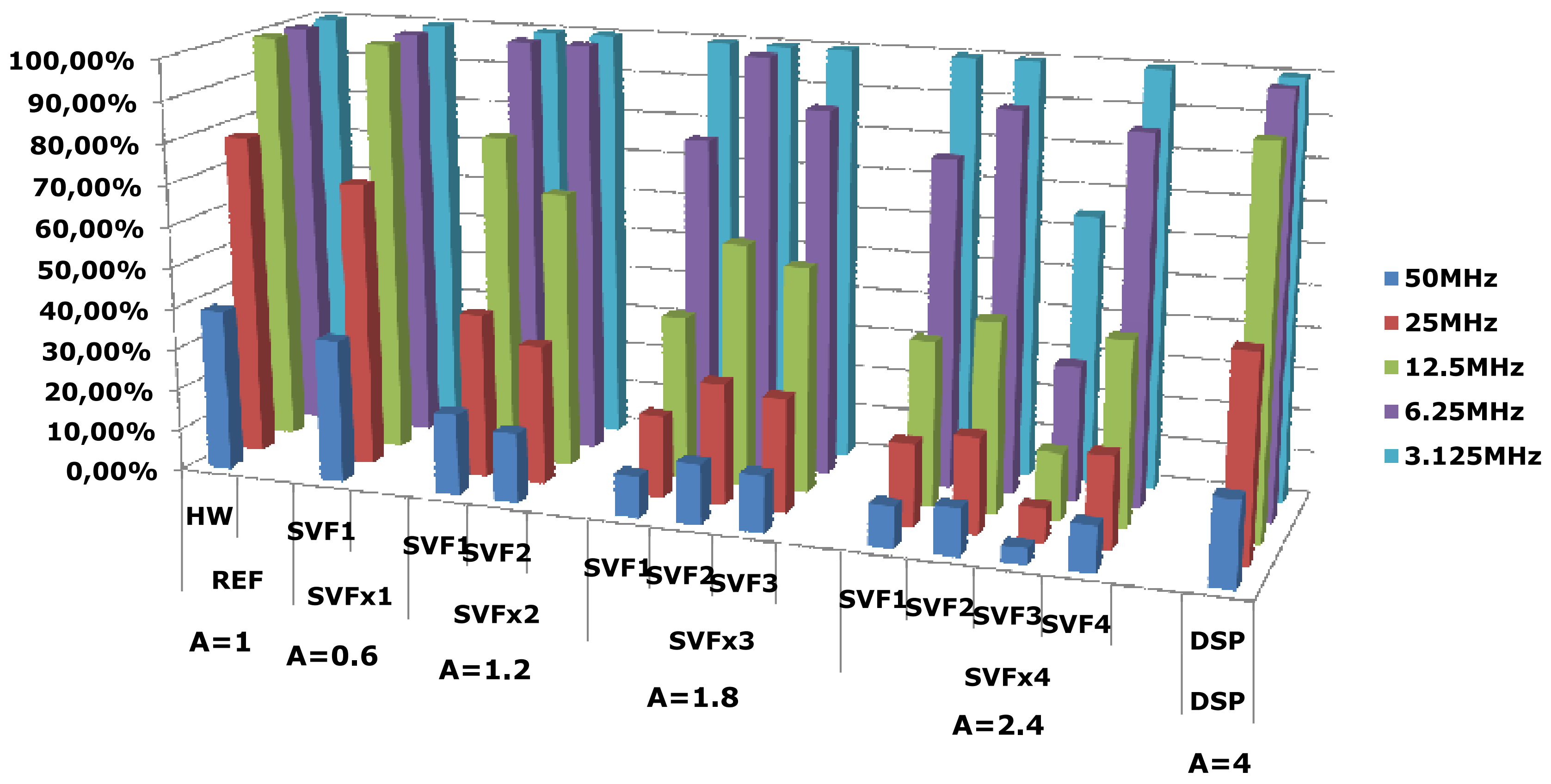


PLATFORMS

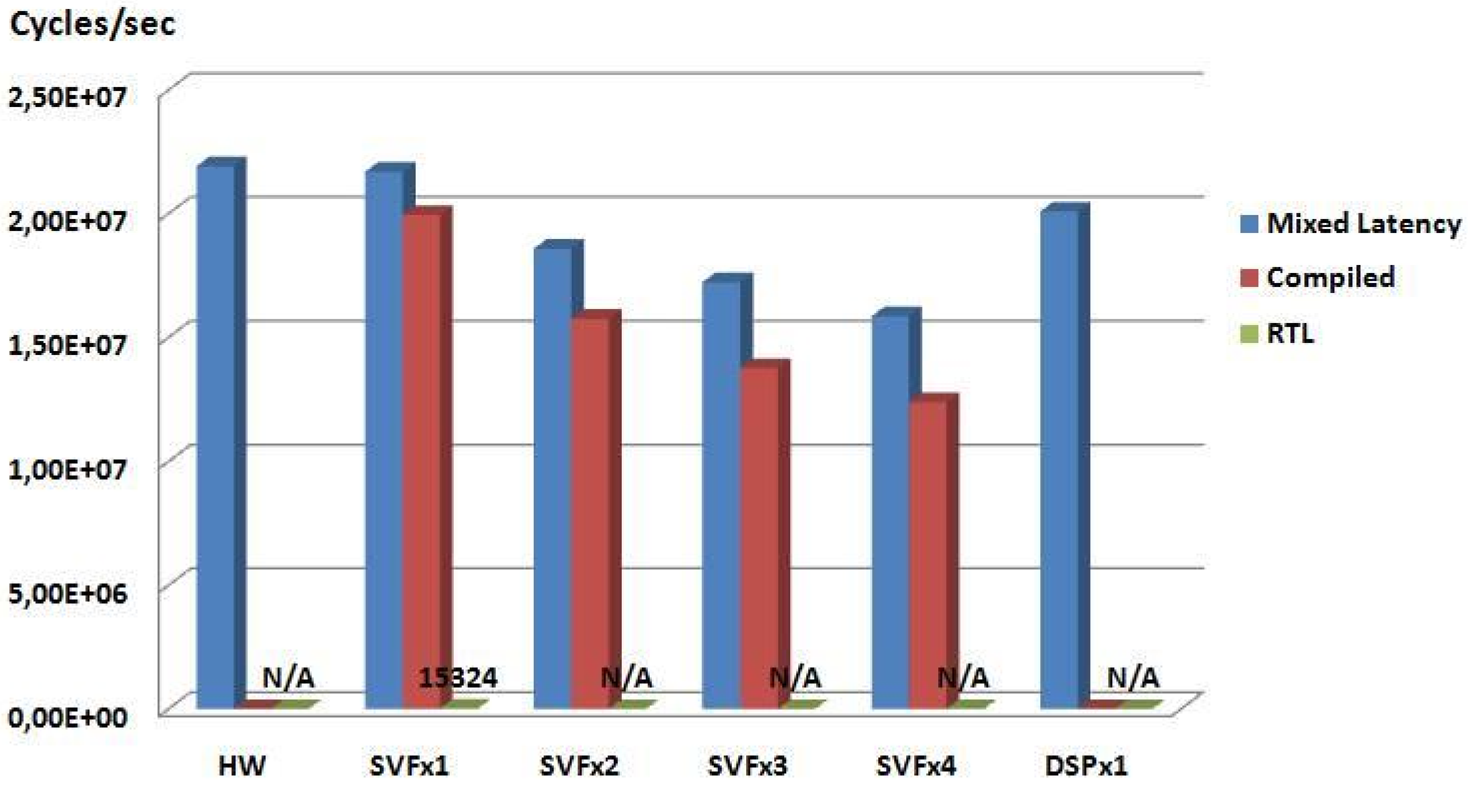
- Available processing elements:
  - Dedicated hardware implementations (HW)
  - Application Specific Instruction-set Processor (SVF)
  - Digital Signal Processor optimized for Audio (DSP)

NAME	DESCRIPTION	RELATIVE AREA
HW (REFERENCE)	Dedicated hardware, ICEpower impl.	1
SVFx1	1 SVF ASIP.	0.6
SVFx2	2 SVF ASIP's.	1.2
SVFx3	3 SVF ASIP's.	1.8
SVFx4	4 SVF ASIP's.	2.4
DSP	1 Audio DSP.	4

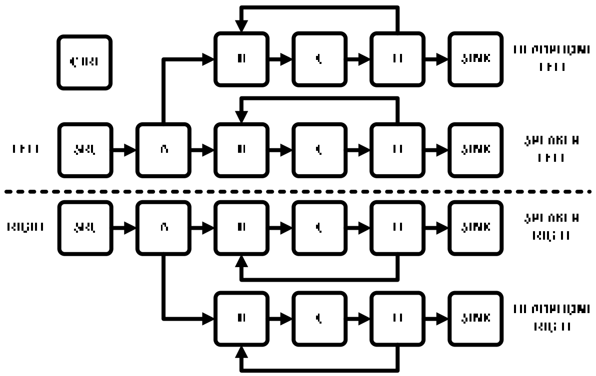
UTILIZATION OF PLATFORMS

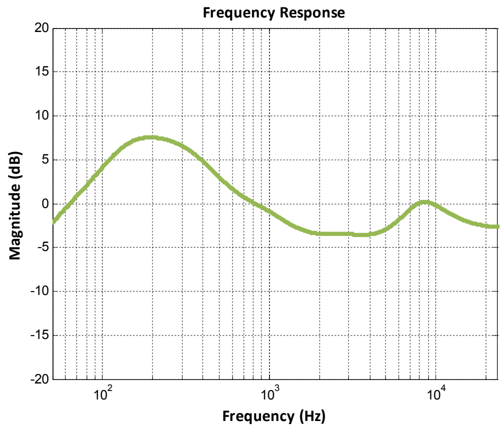


SIMULATION SPEED



ACCURACY





	MIXED LATENCY	COMPILED	RTL
A	84,034	102,034	102,034
B	123,041	129,384	129,384
C	33,011	33,011	33,011
D	153,051	168,056	168,056
TOTAL	393,137	429,143	429,143

OUTLOOK

CONCLUSION & FUTURE WORK

CONCLUSION

- Early performance estimation.
- Flexible construction of models.
- High-level of reusability, using component libraries.
- Support for multiple levels of abstraction to co-exists within a model.
- Excellent refinement possibilities of models.
- High level models to cycle accurate instruction set simulators.

FUTURE WORK

- SystemC / TLM 2.0 integration investigation.
- Better tool support.
  - Specification of models.
  - Specification of applications.
  - "Service Request Compiler"
- In depth case studies – larger and more complex models.
- Investigation of formal analysis possibilities using traditional HCPN methods.