how big can you dream?™

cadence

Platform-Based Co-Design and Co-Development: Experience, Methodology and Trends

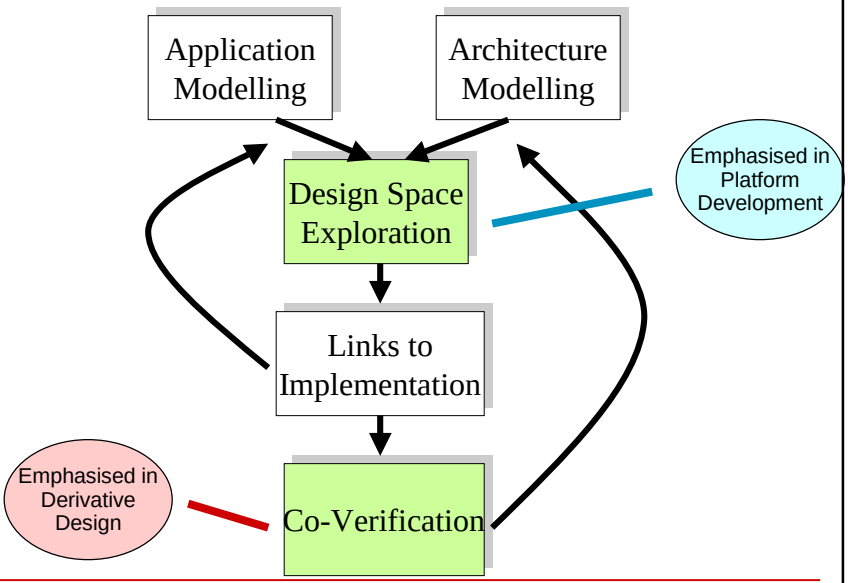
Grant Martin and Jean-Yves Brunel
Cadence Berkeley and Paris
Electronic Design Processes Workshop, Monterey, April 21-23, 2002

Outline

cadence

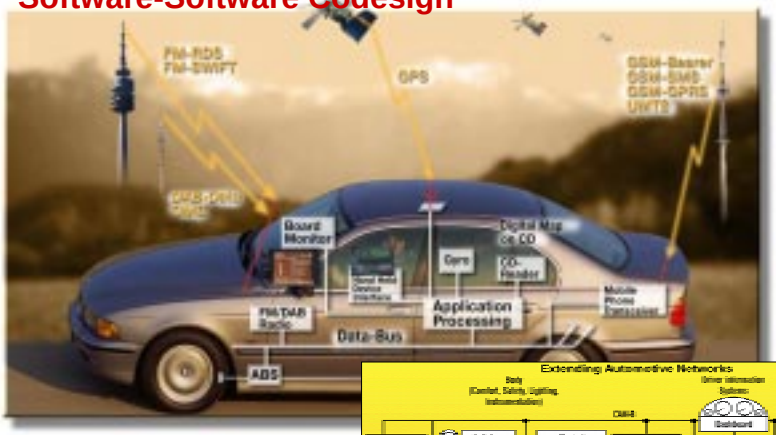
- Design Flows
- Automotive ‘Software-Software’ codesign
- Multimedia Design Space Exploration
- Conclusions

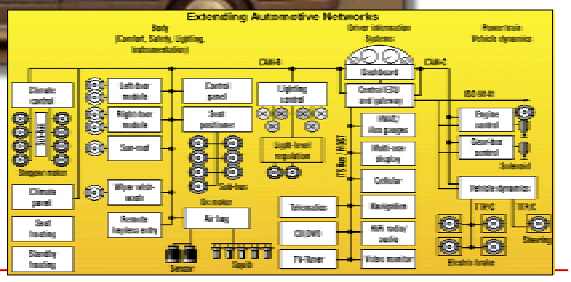
Embedded System Development



3

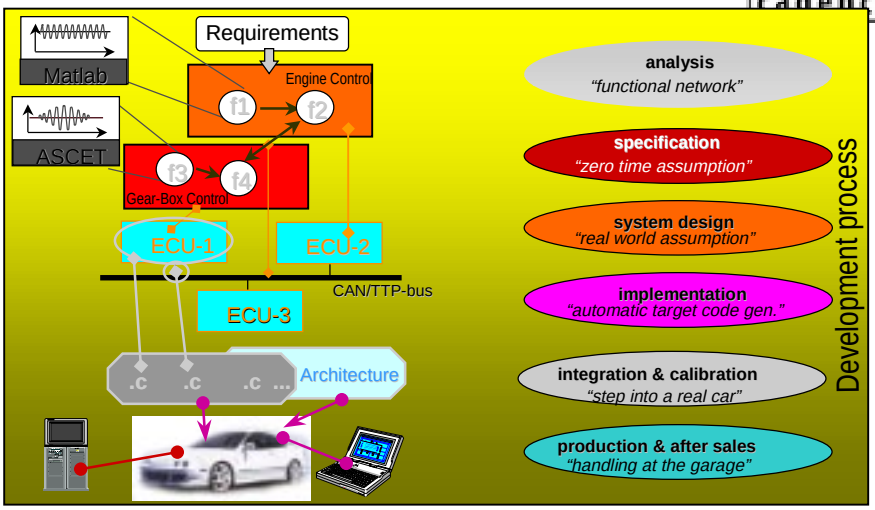
Distributed Automotive Applications over networks – "Software-Software Codesign"





- Electronic Control Units (ECU's)
- Standard buses (TTP, CAN, FlexRay)
- Standard Platforms

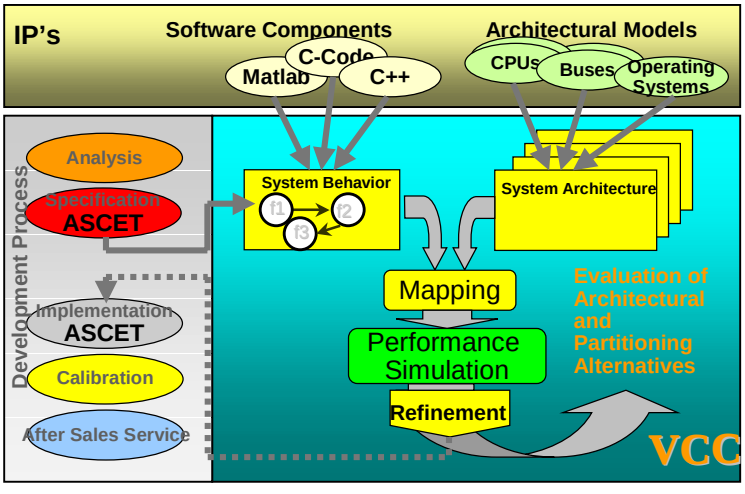
Current Design Practices



- Integration is done too late In the car
- *Tools are PER-ECU – conservative, costly, no tradeoffs*

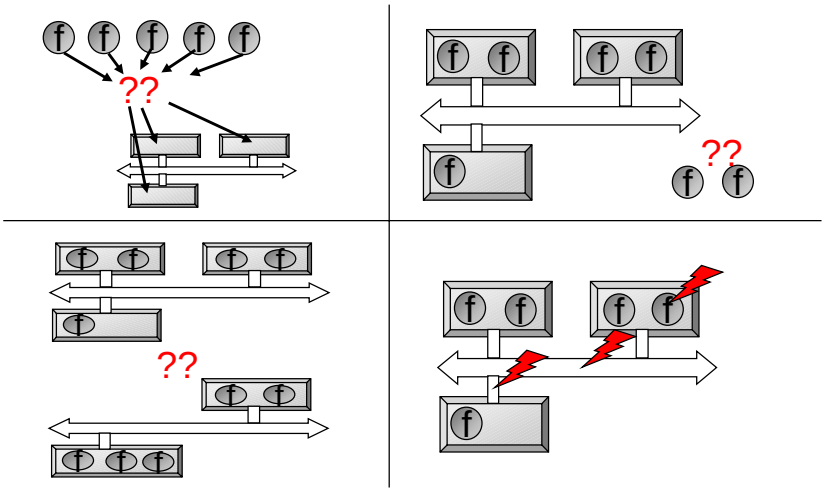
5

Virtual Integration Platform for Distributed Automotive Applications



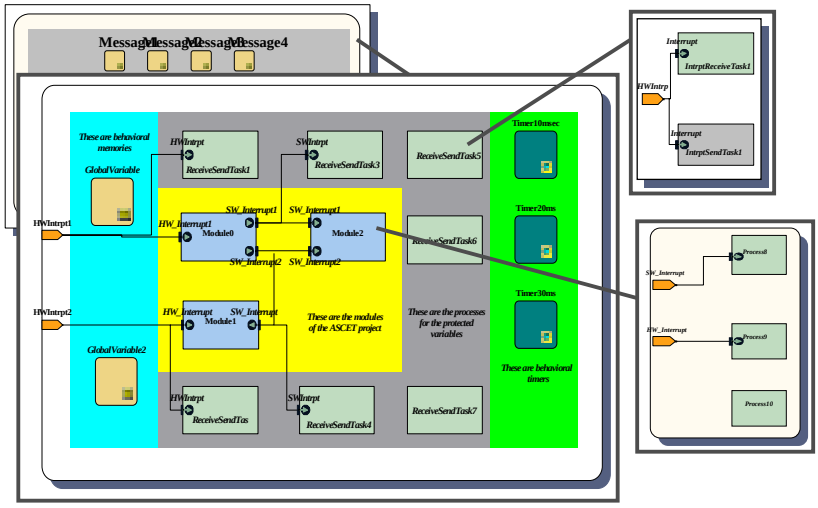
6

Scenarios for SW-driven co-development



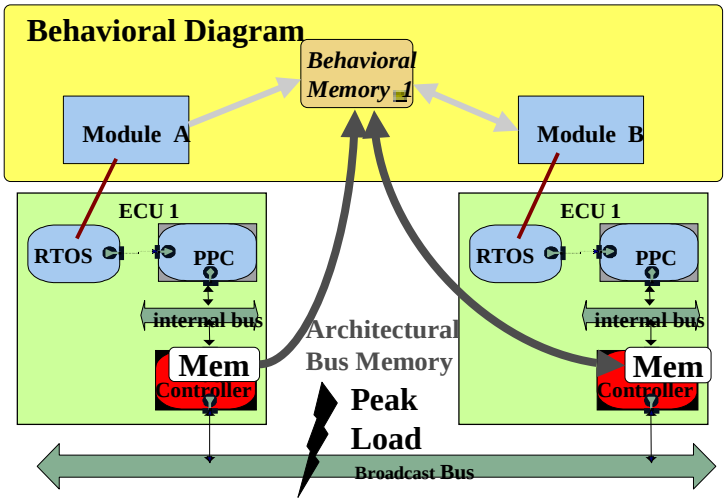
7

ASCET-SD imported project in VCC



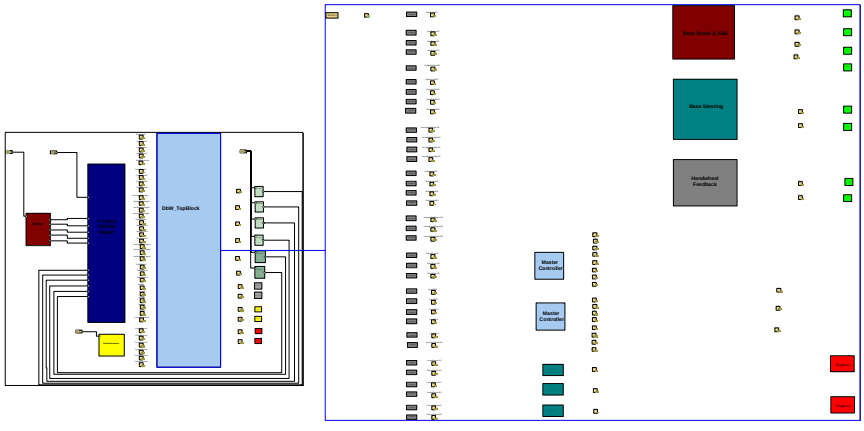
8

Universal Communications Model of Bus



Example Design Flow (1): Power Window

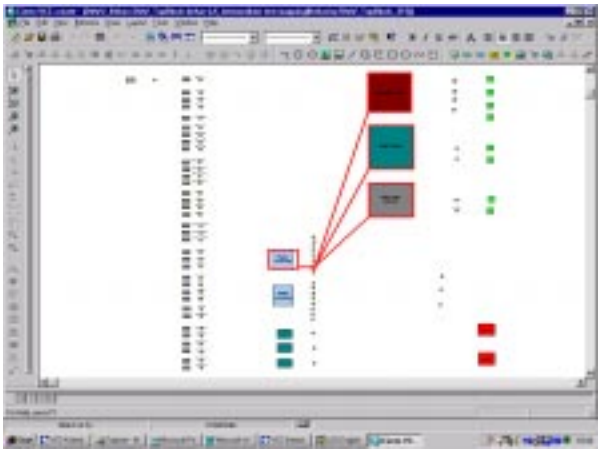
- Definition of a behavioral diagram: Import of functional components (software projects and modules)



10

Design Flow (2)

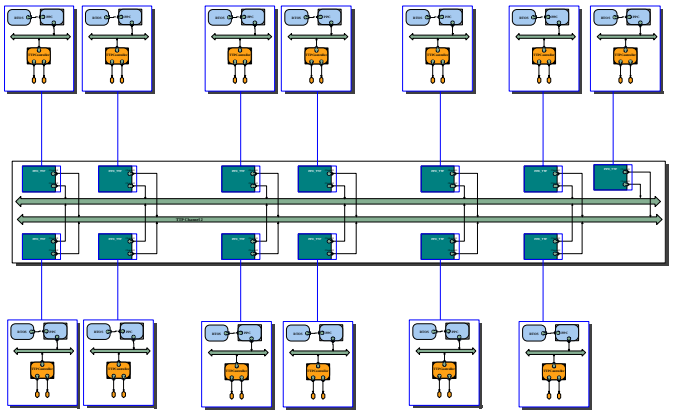
- Generation of an ideal communication between the functional components
  - No delay or error handling considered.
  - ***Functional co-verification***



11

Design Flow (3)

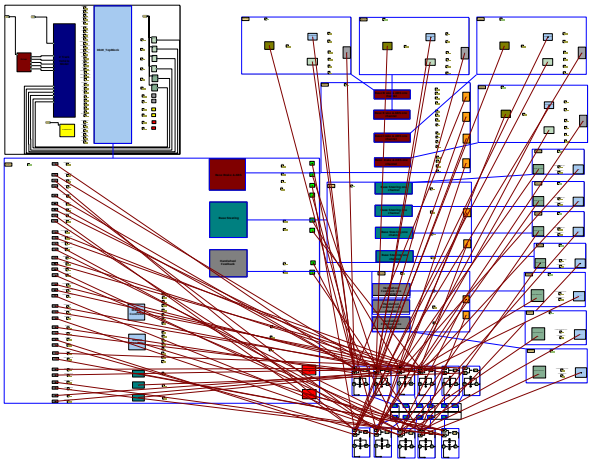
- Creation of an architectural diagram in VCC



12

Design Flow (4)

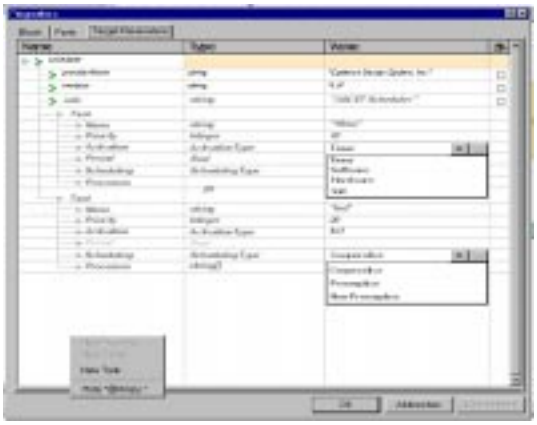
- Mapping the software modules onto the ECU
  - Either retaining the original per-ecu mapping from ASCET-SD or creating a new one

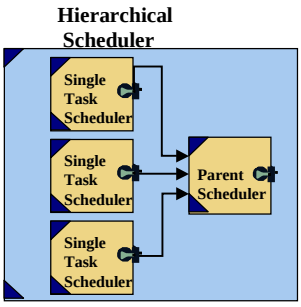


13

Design Flow(5)

- Generation of the CPU scheduling
  - Either manually or automatically in case the original scheduling is preserved

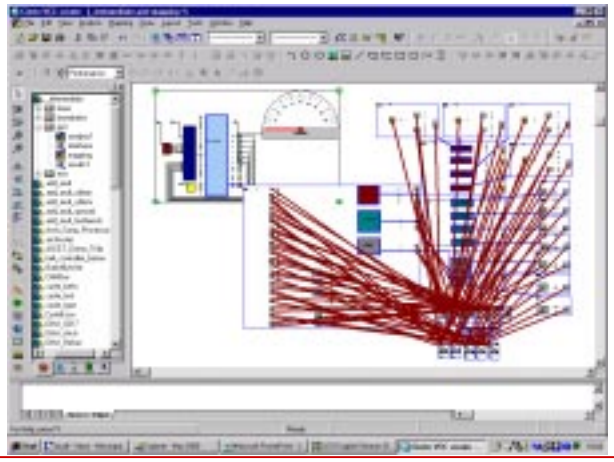




14

Design Flow (6)

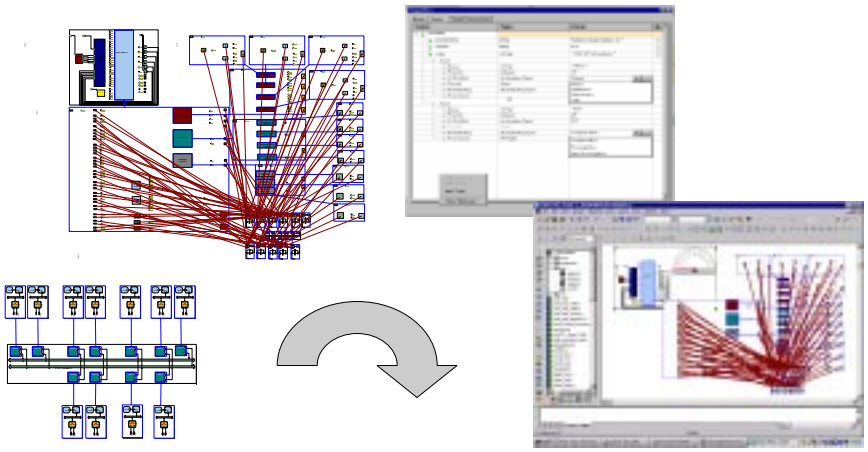
- Computation Performance Simulation
  - No communication performance estimation
  - ***Co-verification of Computational Resource 'fit'***



15

Design Flow(7)

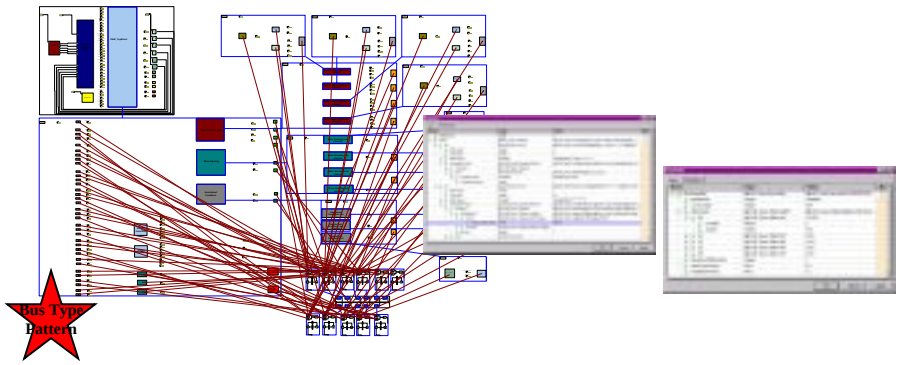
- Design iterations
  - Re-distribution of the functionality and tuning of the scheduling



16

Design Flow(8)

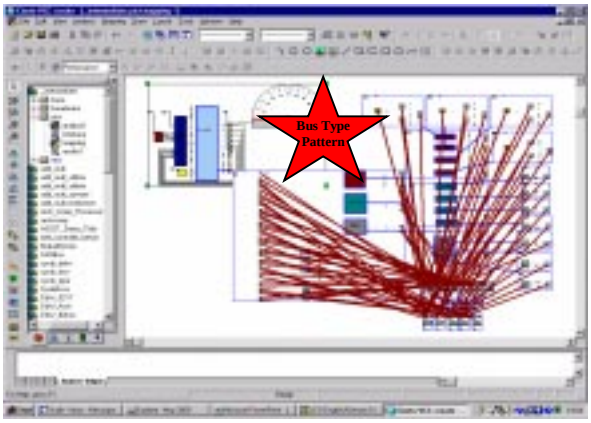
- Initialization of the UCM performance model.
  - Automated generation of an initial communication matrix that carries the dependency of the functional system mapping.
- Definition of a specific bus protocol implementation
  - UCM parameterization. Definition of the communication cycle layout. Data frame definition.



17

Design Flow (9)

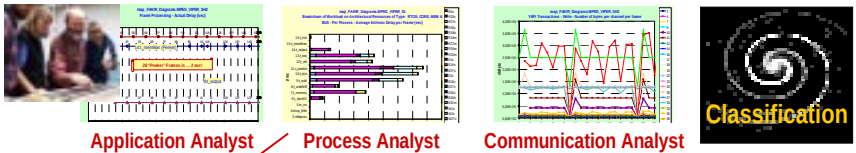
- Performance simulation including the bus latencies
- ***Full System co-verification: both communications and computation***



Design Iterations

18

Multimedia Applications – Design Space Exploration



Application Analyst | Process Analyst | Communication Analyst

Data Analysis Workbooks for specific roles (Excel+StatBox)

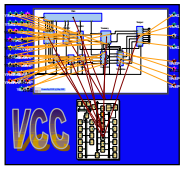
Design Space Navigator (SQL queries)

vccDse Database

vccMap Database

vccSim Database

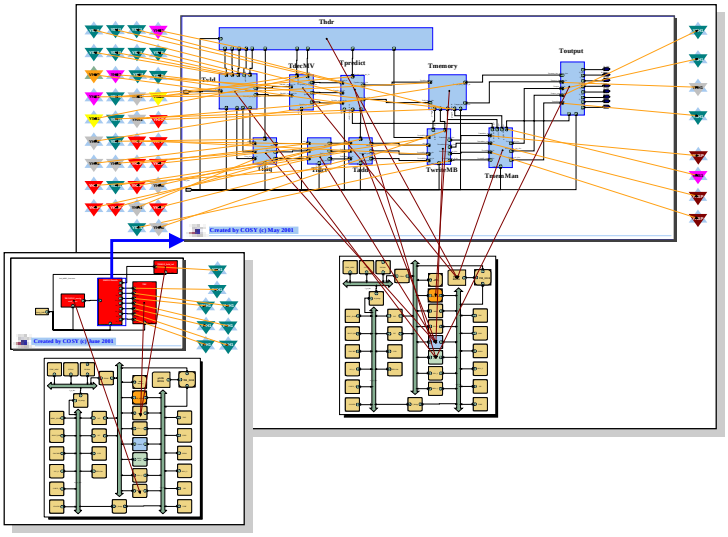
Design Data Exported from VCC diagrams using "VCCAPI"



Performance Data Collected by VCC probes under control of system events

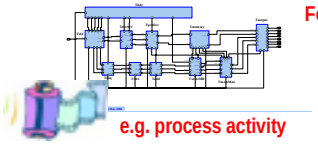
19

Export Mapping Data to DataBase



20

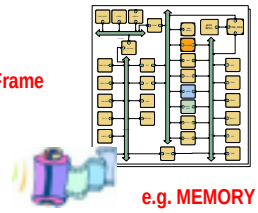
System-Observation-Windows



e.g. process activity

t_vld process

For, e.g. Each MPEG Frame Measure…

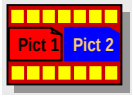


e.g. MEMORY usage

MMI memory

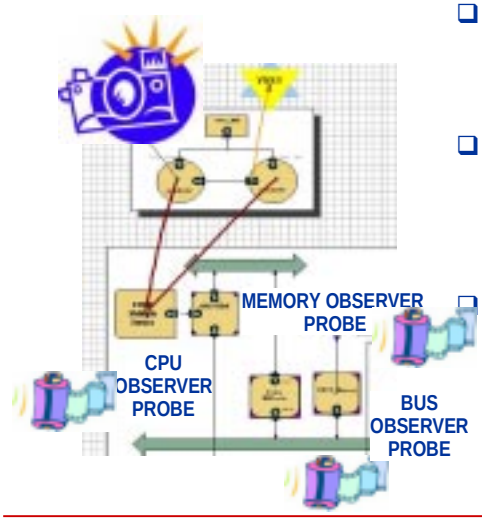
Frame ID	Port Name	Transaction Nb	Item Nb	Actual Delay	Intra Delay
1	Tvld_bits_In	324	20,736	0.14	0.02
1	Tvld_cmd_In	208	208	0.13	0.00
1	Tvld_prop_pic_In	1	1	0.00	0.00
1	Tvld_prop_slice_In	36	36	0.00	0.00
1	mb_QFS_Out	1,622	622,704	6.66	0.65
1	Thdr_status_Out	208	208	0.01	0.01
1	Tisiq_prop_mb_Out	1,622	1,622	0.03	0.03
1	TdecMV_prop_mv_Out	1,622	1,622	0.04	0.04
	…	…	….	…	….
2	Tvld_bits_In	525	33,600	0.23	0.03
2	Tvld_cmd_In	206	206	0.12	0.00
2	Tvld_prop_pic_In	1	1	0.00	0.00
2	Tvld_prop_slice_In	36	36	0.00	0.00
2	mb_QFS_Out	1,619	621,696	6.53	0.65
2	Thdr_status_Out	206	206	0.01	0.01
2	Tisiq_prop_mb_Out	1,619	1,619	0.03	0.03
2	TdecMV_prop_mv_Out	1,619	1,619	0.04	0.04
2	TdecMV_prop_pred_Out	1,619	1,619	0.09	0.09
…	…	…	…	…	…

SOW's



Frame ID	Requestor	Delay Mean	Delay StDev
1	Behavior/in_es_out_sender	5.64E-06	3.00E-06
1	Behavior/decode/t_vld_Tvld_bits_In_receiver	2.91E-06	1.20E-06
1	Behavior/decode/t_hdr_Tvld_cmd_Out_sender	2.40E-07	1.17E-14
1	Behavior/decode/t_vld_Tvld_cmd_In_receiver	2.40E-07	1.17E-14
1	Behavior/decode/t_hdr_Tvld_prop_pic_Out_sender	2.16E-06	0.00E+00
1	Behavior/decode/t_vld_Tvld_prop_pic_In_receiver	2.16E-06	0.00E+00
1	Behavior/decode/t_hdr_Tvld_prop_slice_Out_sender	7.20E-07	1.22E-14
….	….	…	…
2	Behavior/in_es_out_sender	5.64E-06	3.00E-06
2	Behavior/decode/t_vld_Tvld_bits_In_receiver	2.91E-06	1.20E-06
2	Behavior/decode/t_hdr_Tvld_cmd_Out_sender	2.40E-07	1.17E-14
2	Behavior/decode/t_vld_Tvld_cmd_In_receiver	2.40E-07	1.17E-14
2	Behavior/decode/t_hdr_Tvld_prop_pic_Out_sender	2.16E-06	0.00E+00
2	Behavior/decode/t_vld_Tvld_prop_pic_In_receiver	2.16E-06	0.00E+00
2	Behavior/decode/t_hdr_Tvld_prop_slice_Out_sender	7.20E-07	1.22E-14
2	Behavior/decode/t_vld_Tvld_prop_slice_In_receiver	7.20E-07	1.22E-14
…	…	…	…

"Probe-Synch" & Observer Probes



- Probe-Synch is triggered on conditions in a behavioral block (i.e. MPEG frame decoded)
- Control up to 200 distributed observer probes of different types: i.e Memory probes, Bus probes, CPU "Delay" probes etc…
- Observer Probes record summary data at the granularity defined by the peeker

Queries 1: link Map & Simulation data

```
Calculate Average Actual & Intrinsic Communication Rates of all "YAPI"
application level Channels
SELECT simComAppYapi.sessionCounter AS frameID...,
simComAppYapi.actualDelay,...
FROM
mapComAppYapi INNER JOIN fctProc ON mapComAppYapi.srcProcID =
fctProc.ID...
WHERE
simComAppYapi.VccInstanceName)=[fctProc].[diagName] & "/" &
[fctProc].[procName];
```

chanID	frameID	nbTransaction	nbItem	nbByte	actualDelay	intrinsicDelay
2	3	5,76E+02	5,76E+02	6,91E+03	6,06E-04	6,06E-04
2	10	5,10E+01	5,10E+01	6,12E+02	4,56E-05	4,56E-05
16	18	1,62E+03	1,62E+03	4,54E+04	7,26E-02	3,54E-05

Simulation "Frame" Context

Simulation results

Key to retrieve Design "Mapping" Decision

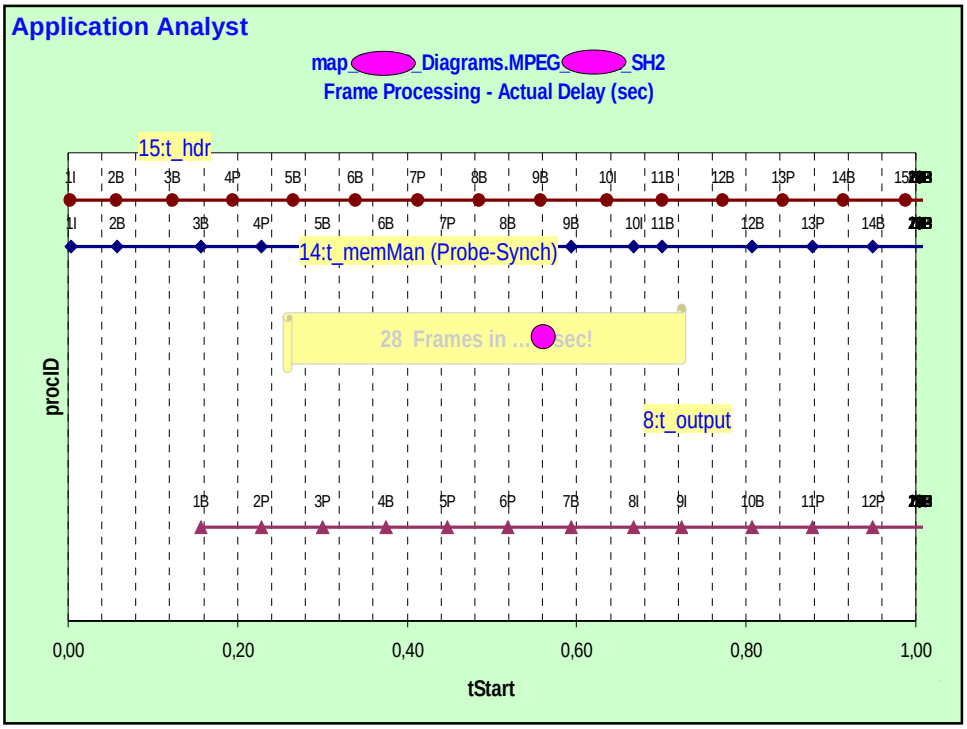
23

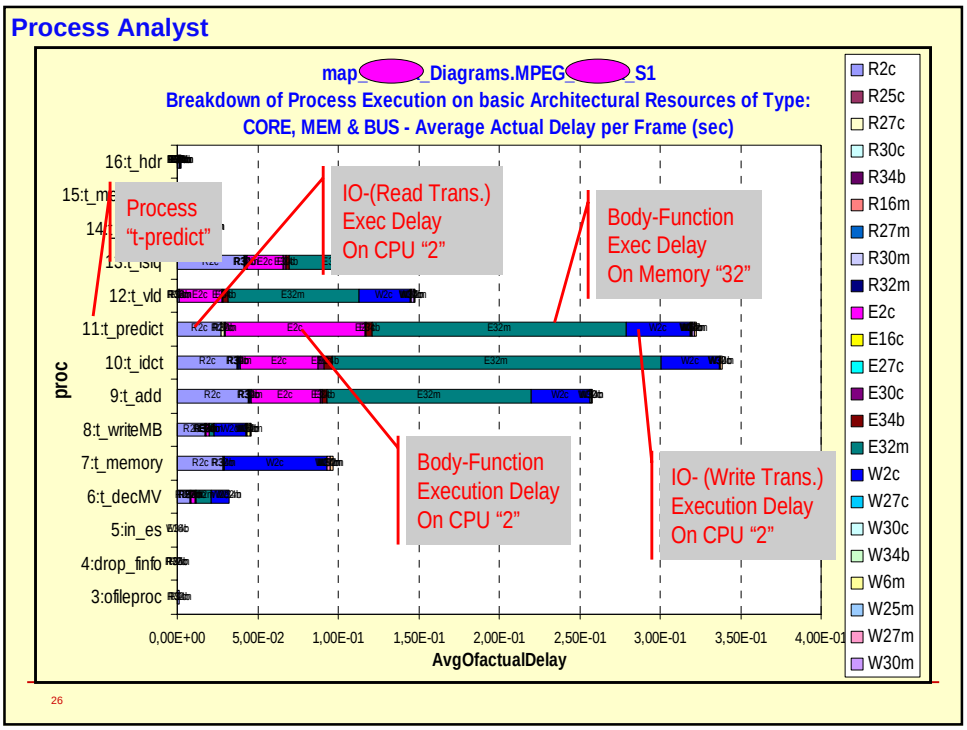
Queries 2: calculate basic Statistics

```
Calculate Average Actual & Intrinsic Communication Rates of all
"YAPI" application level Channels
  SELECT DISTINCTROW
  Avg([nbByte]/[actualDelay]) AS AvgOfActualRate,
  StDev([nbByte]/[actualDelay]) AS StDevOfActualRate,...
  Avg([nbByte]/[intraDelay]) AS AvgOfIntraRate, ... INTO
  staComAppYapiRate_perChan
  FROM mkStaComAppYapiRate_perChan_step1
  GROUP BY chanID...;
```

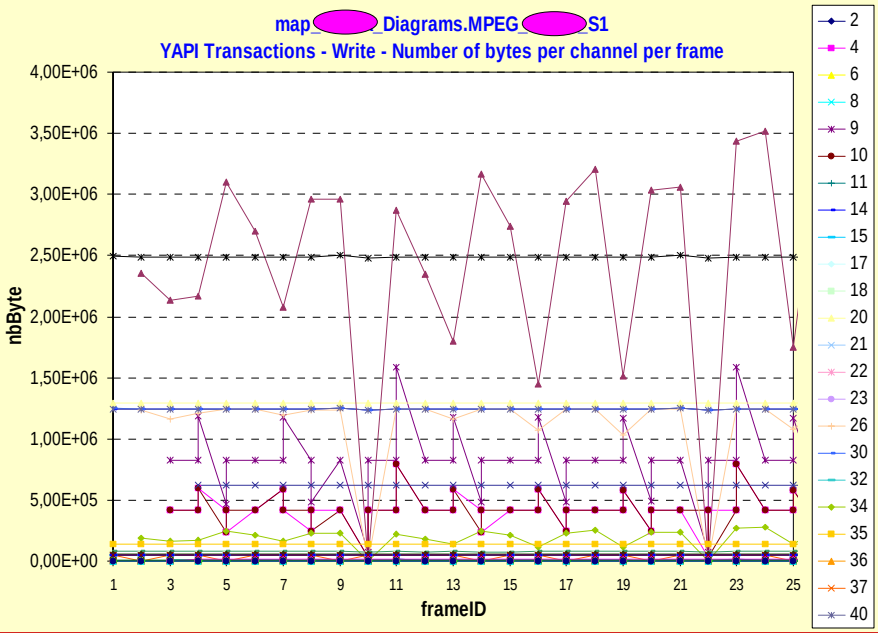
comAppClass	transaction	chanID	portID	AvgOfActualRate	StDevOfActualRate	MinOfActualRate	MaxOfActualRate	AvgOfIntraRate
YAPI	Rd	2	14	5,61E+06	5,98E+04	5,55E+06	5,77E+06	6,09E+06
YAPI	Wr	2	48	7,34E+06	1,58E+06	6,47E+06	1,21E+07	1,13E+07
YAPI	Rd	3	15	6,27E+07		6,27E+	6,27E+0	6,67E+07

24


Application Analyst
map_Diagrams.MPEG_SH2
Frame Processing - Actual Delay (sec)
15:t_hdr
14:t_memMan (Probe-Synch)
28 Frames in ... sec!
8:t_output
procID
0,00
0,20
0,40
0,60
0,80
1,00
tStart

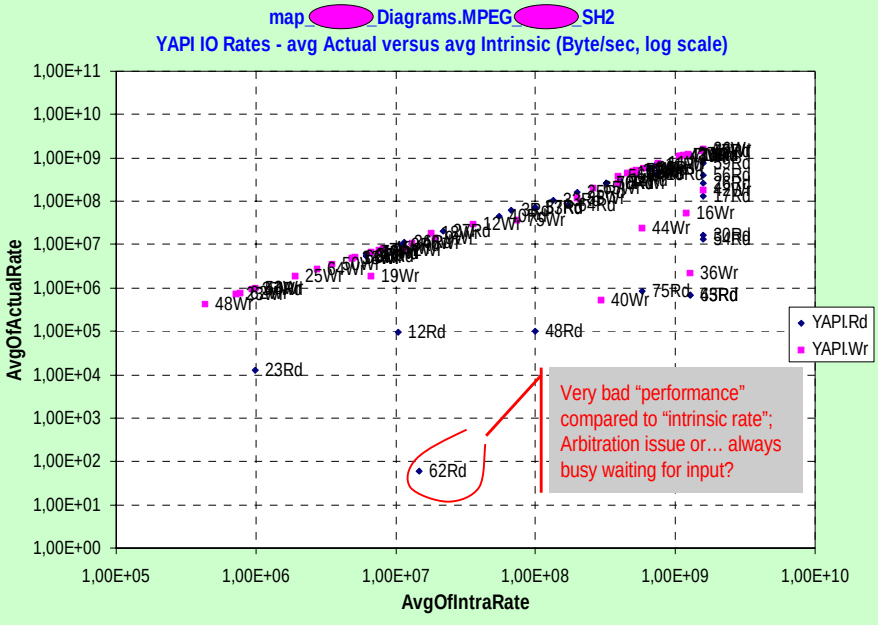

Process Analyst
map_Diagrams.MPEG_S1
Breakdown of Process Execution on basic Architectural Resources of Type:
CORE, MEM & BUS - Average Actual Delay per Frame (sec)
Process "t-predict"
IO-(Read Trans.) Exec Delay On CPU "2"
Body-Function Exec Delay On Memory "32"
Body-Function Execution Delay On CPU "2"
IO- (Write Trans.) Execution Delay On CPU "2"
16:t_hdr
12:t_vld
11:t_predict
10:t_idct
9:t_add
8:t_writeMB
7:t_memory
6:t_decMV
5:in_es
4:drop_finfo
3:ofileproc
proc
0,00E+00
5,00E-02
1,00E-01
1,50E-01
2,00E-01
2,50E-01
3,00E-01
3,50E-01
4,00E-01
AvgOfactualDelay
R2c
R25c
R27c
R30c
R34b
R16m
R27m
R30m
R32m
E2c
E16c
E27c
E30c
E34b
E32m
W2c
W27c
W30c
W34b
W6m
W25m
W27m
W30m
26

Communication Analyst

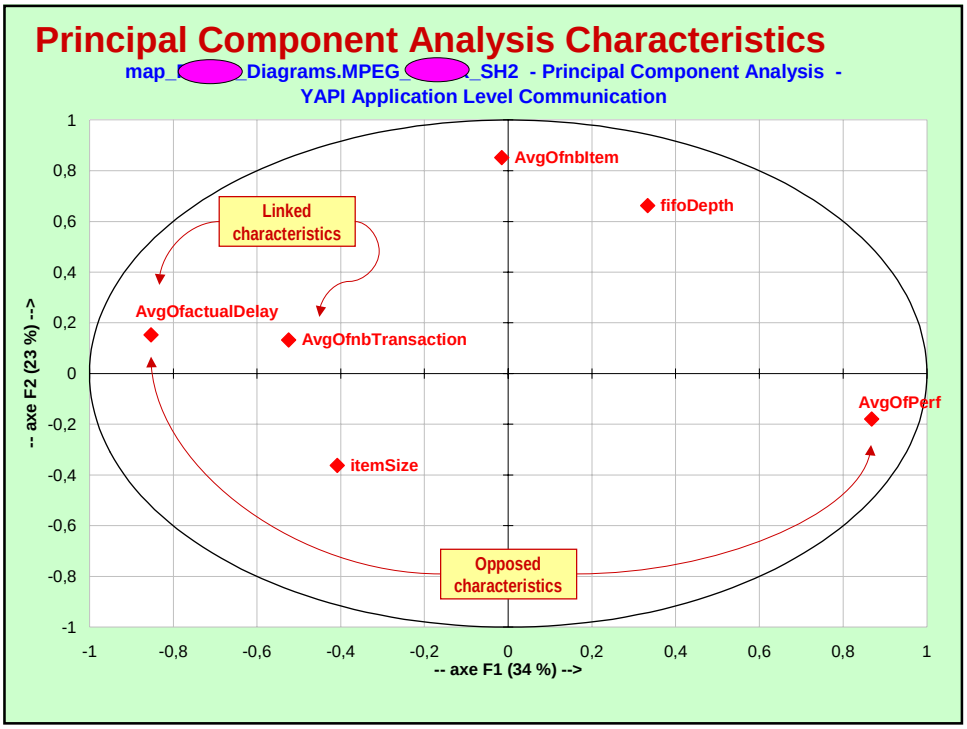


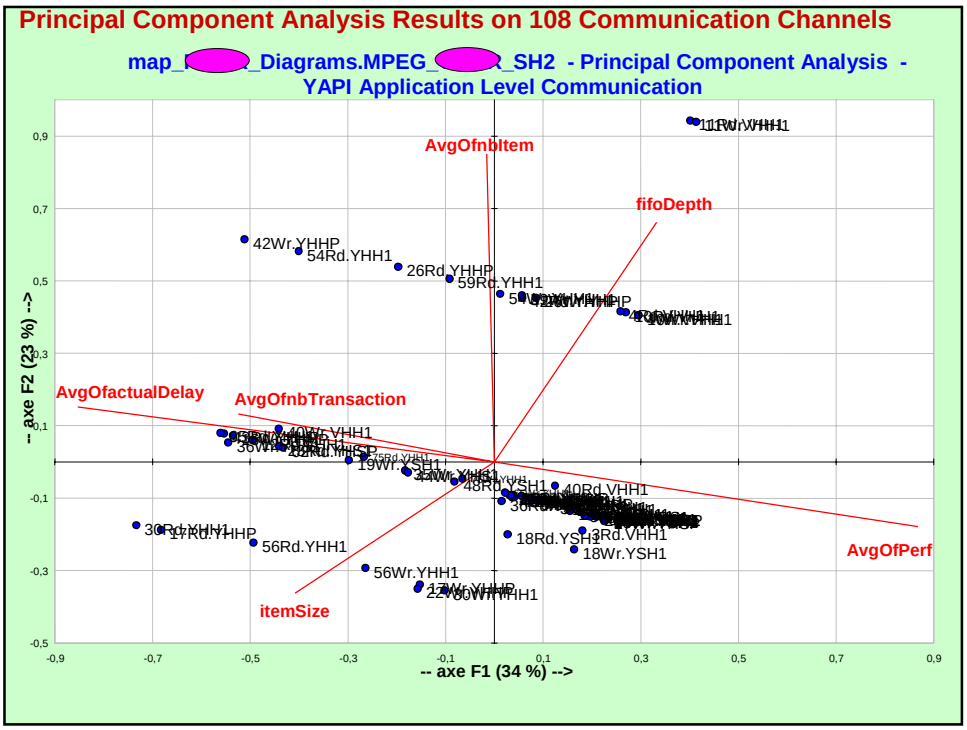
27

Communication Analyst

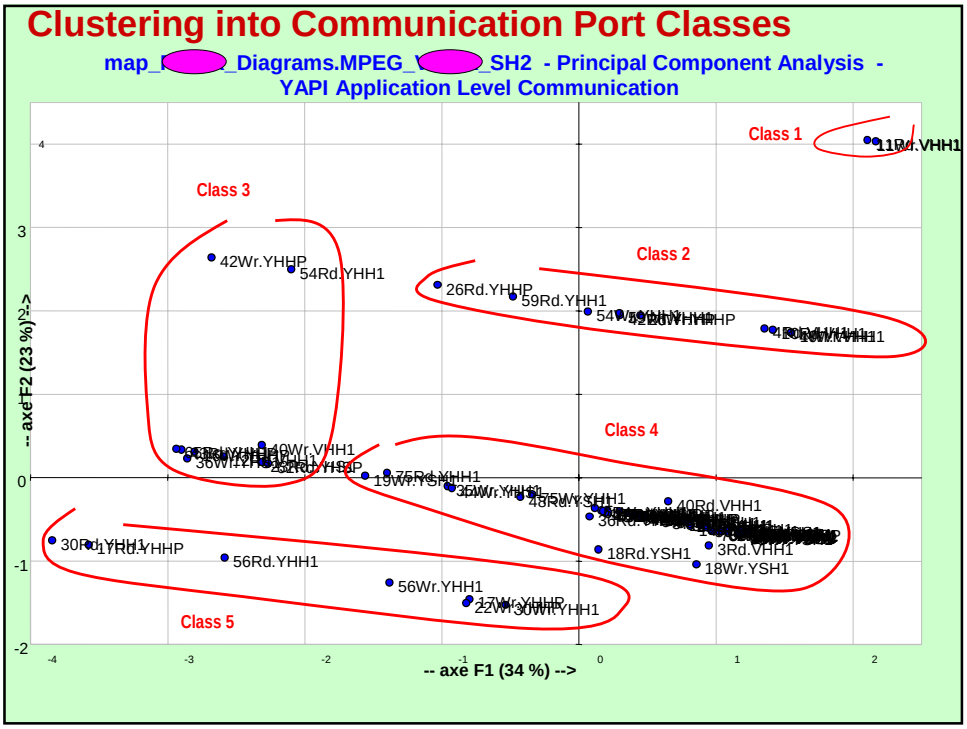


28


Principal Component Analysis Characteristics
map_[redacted]_Diagrams.MPEG_[redacted]_SH2 - Principal Component Analysis - YAPI Application Level Communication
AvgOfnbItem
fifoDepth
Linked characteristics
AvgOfactualDelay
AvgOfnbTransaction
AvgOfPerf
itemSize
Opposed characteristics
-- axe F2 (23 %) -->
-- axe F1 (34 %) -->
1
0,8
0,6
0,4
0,2
0
-0,2
-0,4
-0,6
-0,8
-1
-1
-0,8
-0,6
-0,4
-0,2
0
0,2
0,4
0,6
0,8
1


Principal Component Analysis Results on 108 Communication Channels
map_[redacted]_Diagrams.MPEG_[redacted]_SH2 - Principal Component Analysis - YAPI Application Level Communication
AvgOfnbItem
fifoDepth
AvgOfactualDelay
AvgOfnbTransaction
AvgOfPerf
itemSize
42Wr.YHHP
54Rd.YHH1
26Rd.YHHP
59Rd.YHH1
40Rd.VHH1
18Rd.YSH1
18Wr.YSH1
56Rd.YHH1
56Wr.YHH1
-- axe F2 (23 %) -->
-- axe F1 (34 %) -->
0,9
0,7
0,5
0,3
0,1
-0,1
-0,3
-0,5
-0,9
-0,7
-0,5
-0,3
-0,1
0,1
0,3
0,5
0,7
0,9

Clustering into Communication Port Classes



Conclusions

- Platform-based design is a reality in several industries
  - Particular applications drive particular methodologies
- Fixed, SW-driven platforms require co-verification approaches for derivatives
  - Automotive example
- Platform creation can make use of effective design space exploration methods
  - Multimedia example
- In future, new platform architectures and new co-development methods may be ‘co-developed’ to allow effective exploitation of architectural features