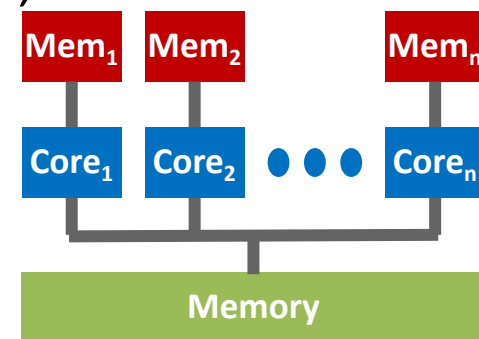
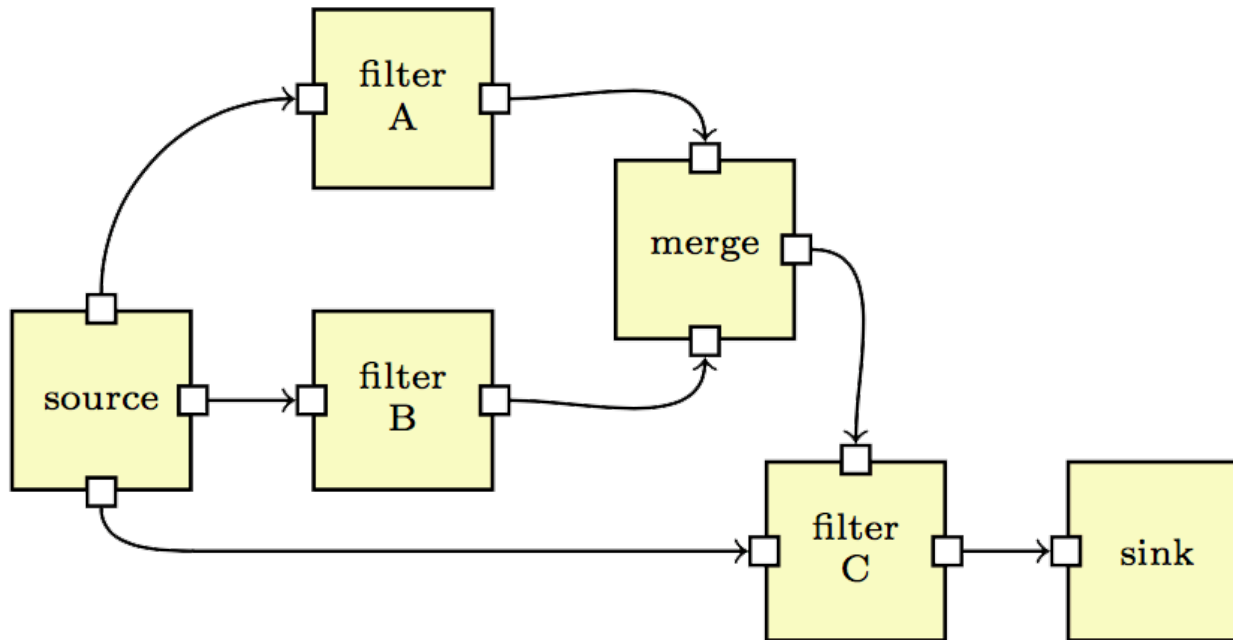


- Exascale Computing
 - Shared Memory + Distributed Memory =
 - Clustered Memory Architectures
 - Kalray MPPA, Parallella, Shared i3, i5, i7 big.LITTLE, GPU ...
- Low Power High Performance Computing
 - Heterogenous Processing Elements
 - FPGA / GPU / Kalray MPPA Coprocessors
- How to choose the « best » architecture ?
 - Processing Elements + Interconnect + Memories
- How to program it ?
 - OpenCL, OpenMP, CUDA, etc. are ALL Hardware specific
- What if hardware upgrades ?
 - Quantum computing opportunities ?!



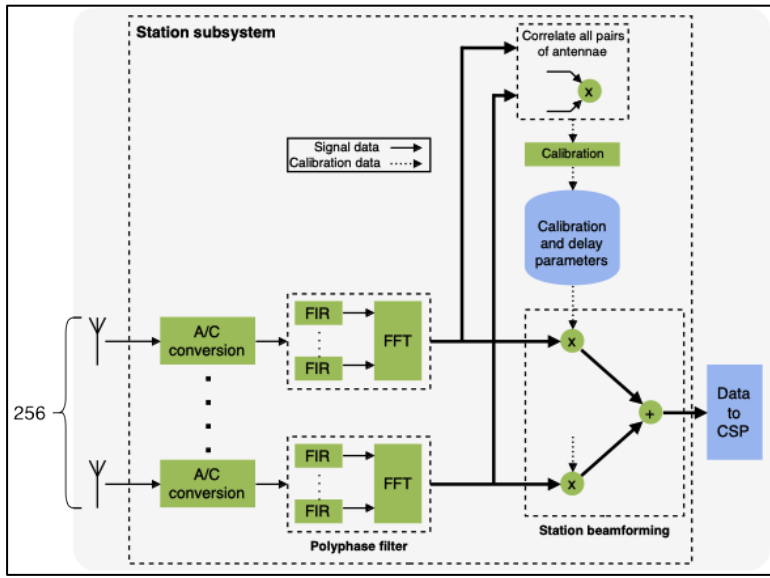
- Algorithm's Parameters exploration
 - Ill-posed problem → iterative algorithms
 - How to choose the number of iterations (Minor / Major loops) ?
 - Computing power constraints !
- Algorithms / Software updates
 - Direction Independant → Direction Dependant Algorithms



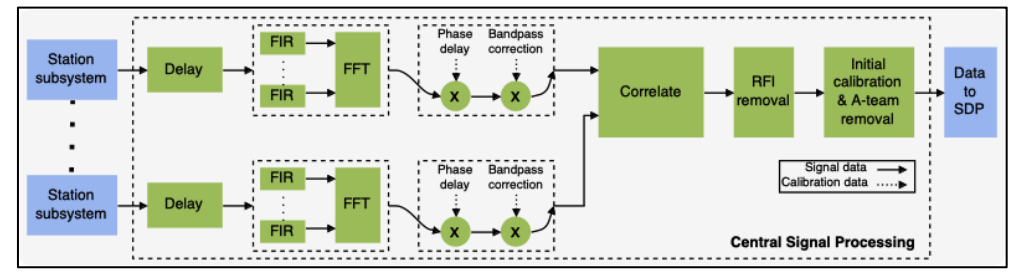
Kahn, 1974 : Kahn Process Network

SKA is a Dataflow Pipeline !

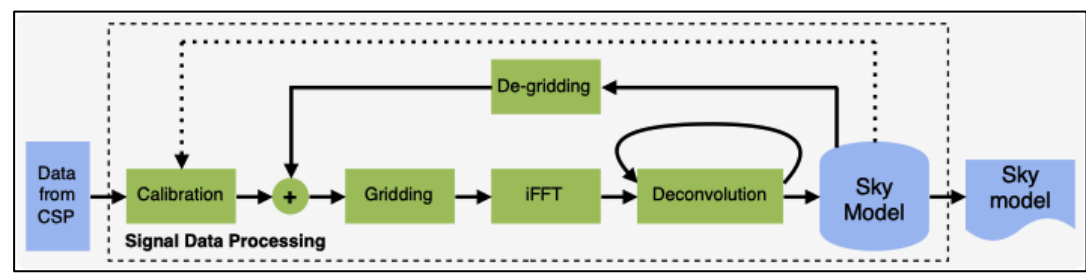
SKA Station



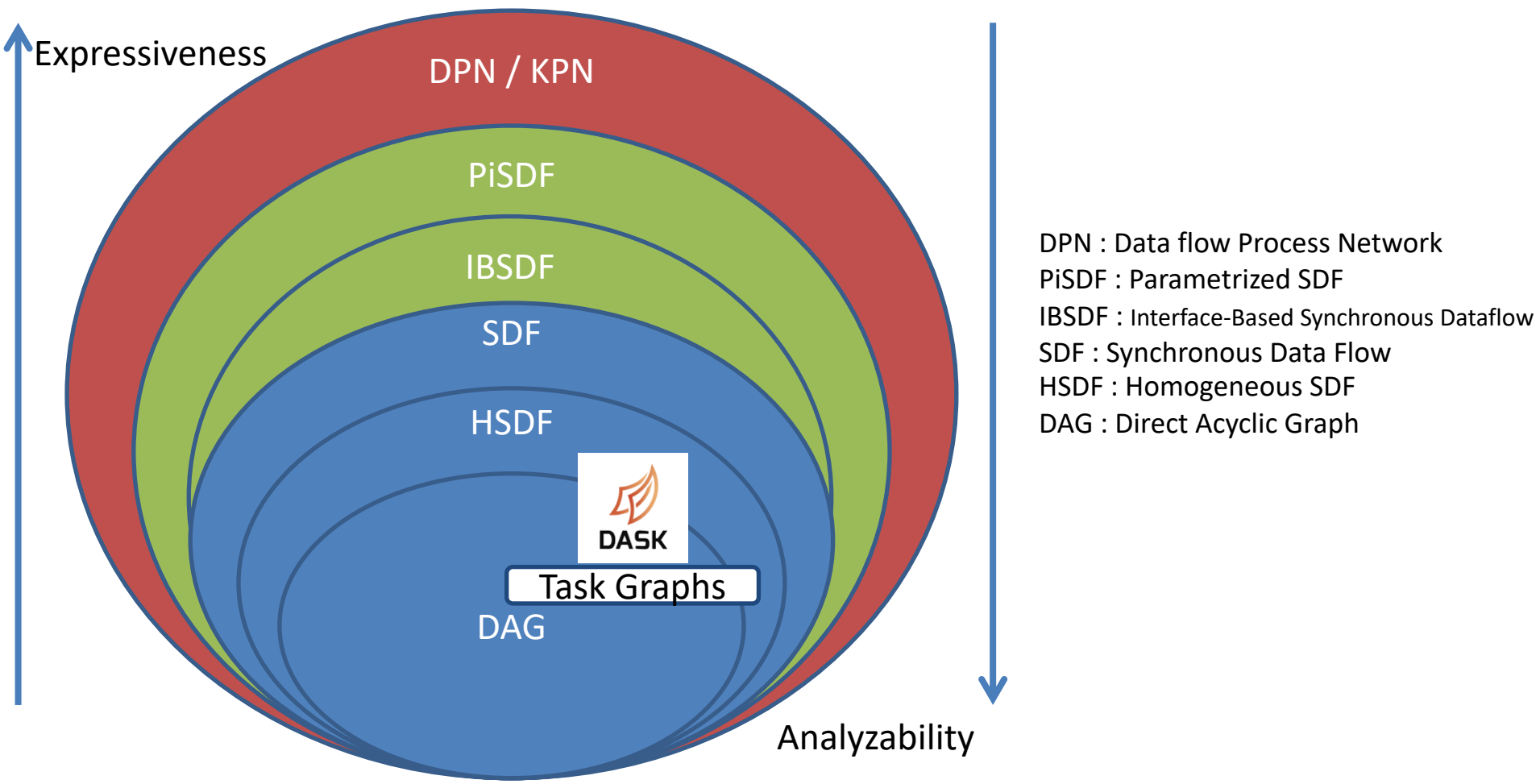
SKA Central Signal Processing (CSP)



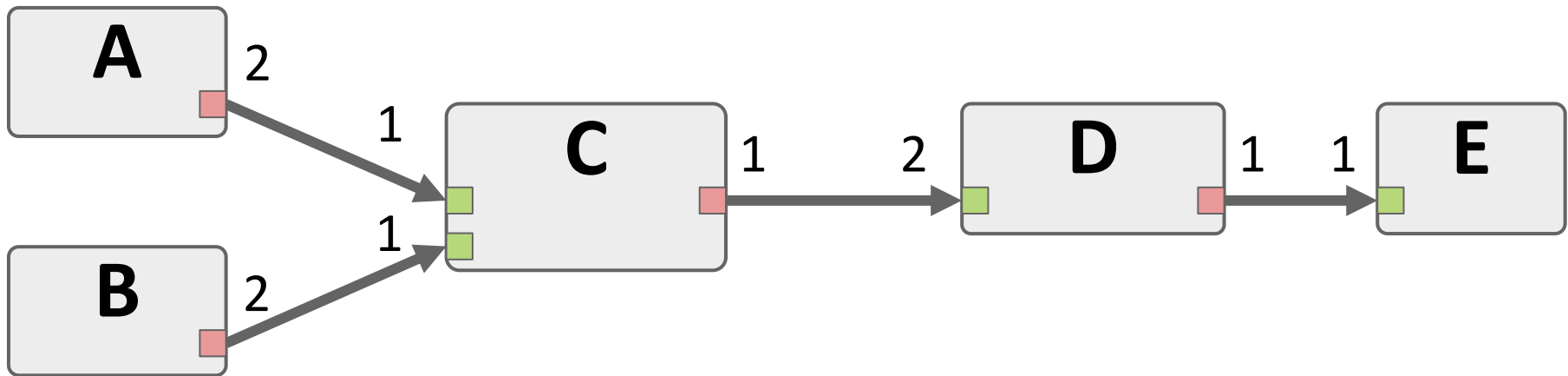
SKA Science Data Processing (SDP)



- Dataflow models / Semantics / MoCs (Model of Computation)

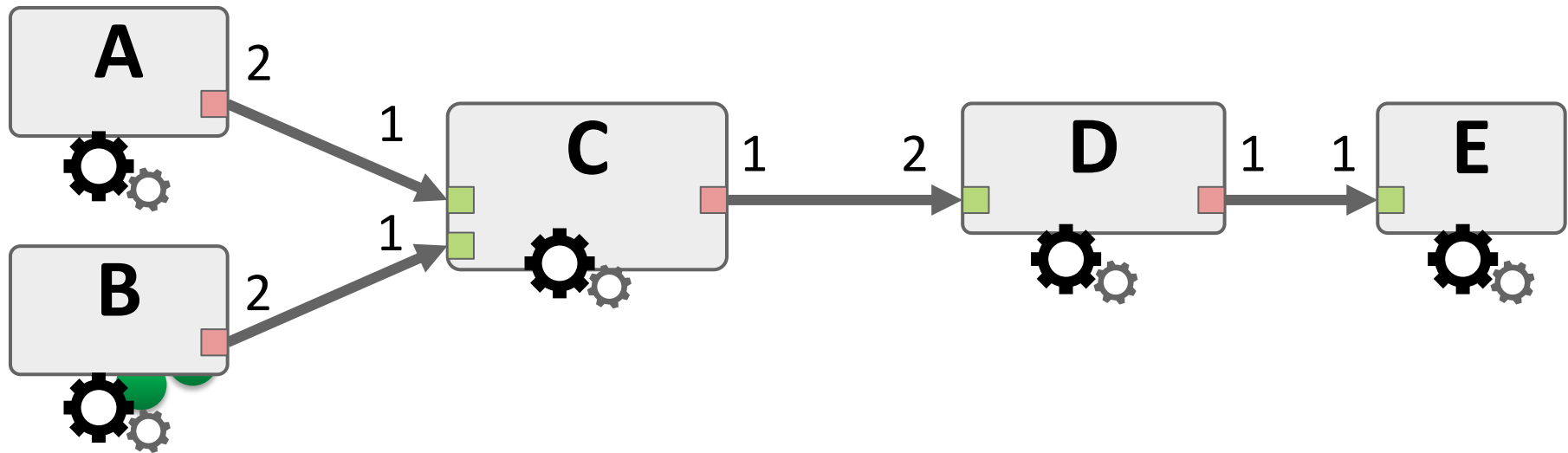


- Actors and Data Ports
- FIFO Queues

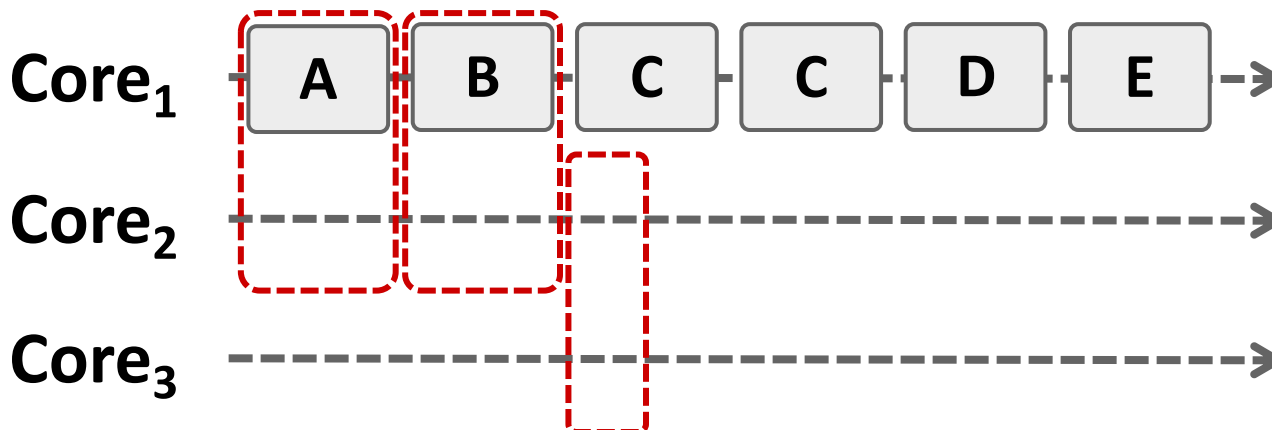
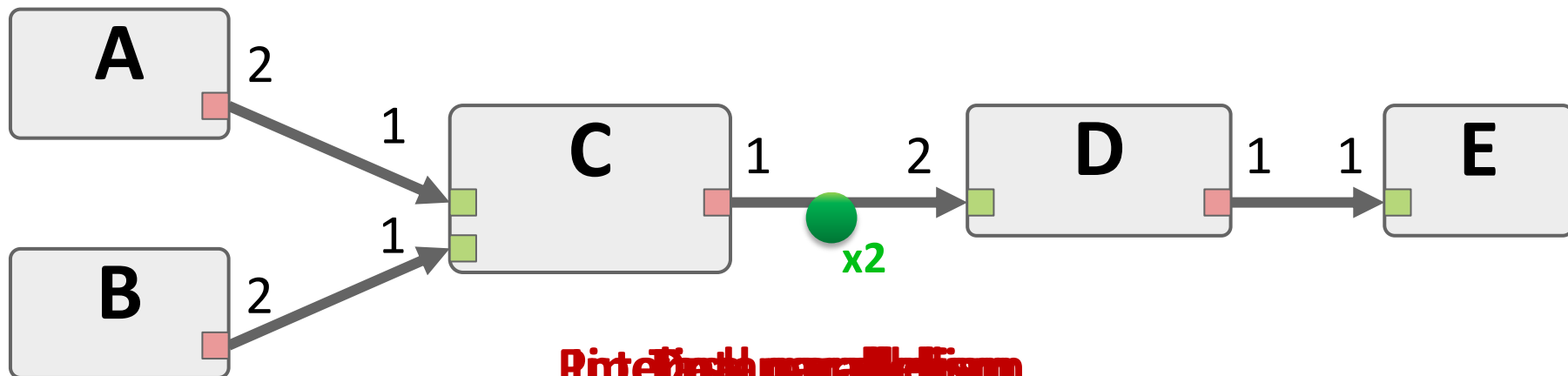


[LM87a] E. Lee and D. Messerschmitt, "Synchronous data flow",
Proceedings of the IEEE, 1987.

- **Data-driven execution**
 - An actor is fired when its input FIFOs contain enough data-tokens.

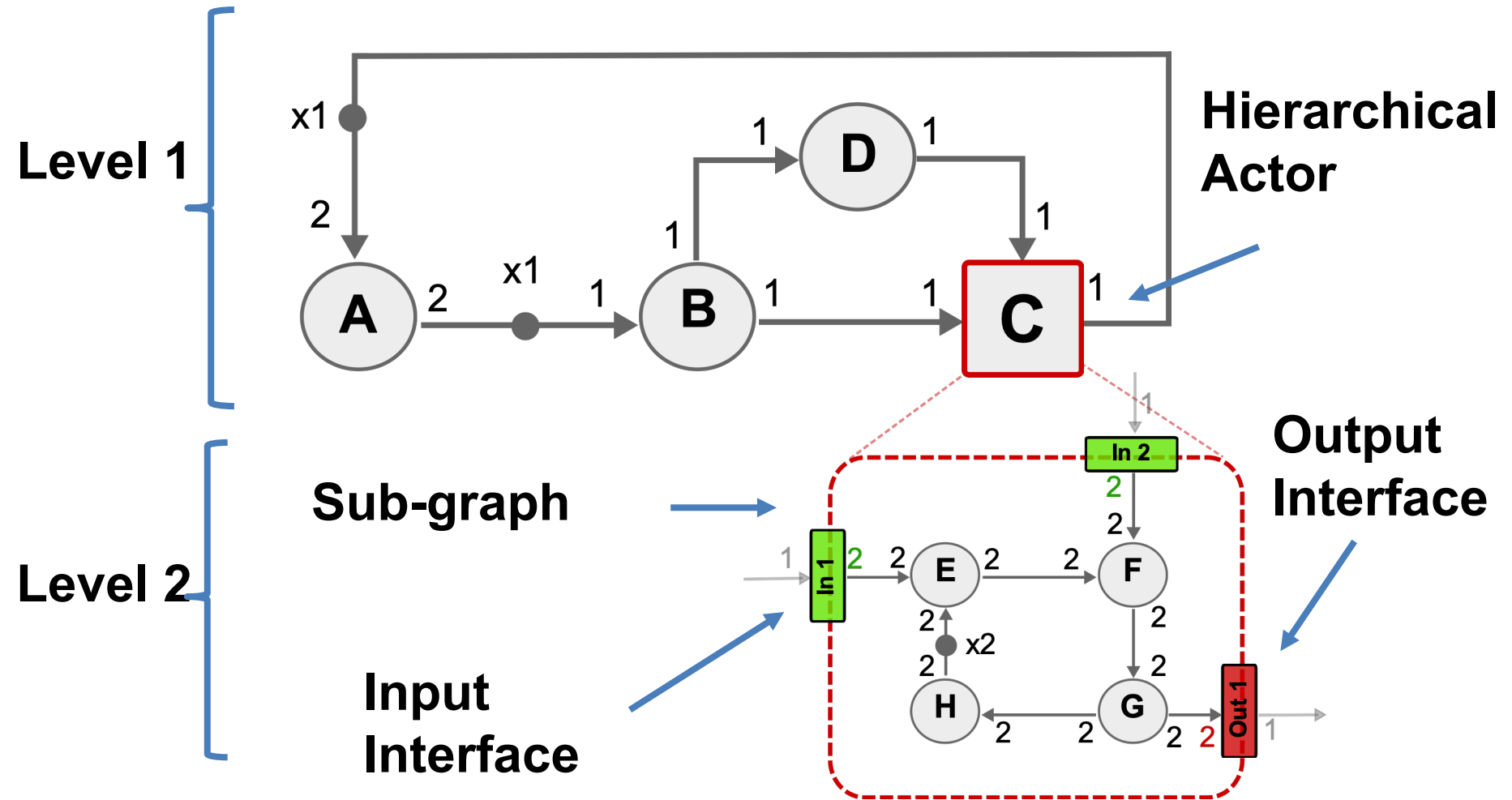


- Expression of parallelisms:
 - Task / Data/ Pipeline Internal

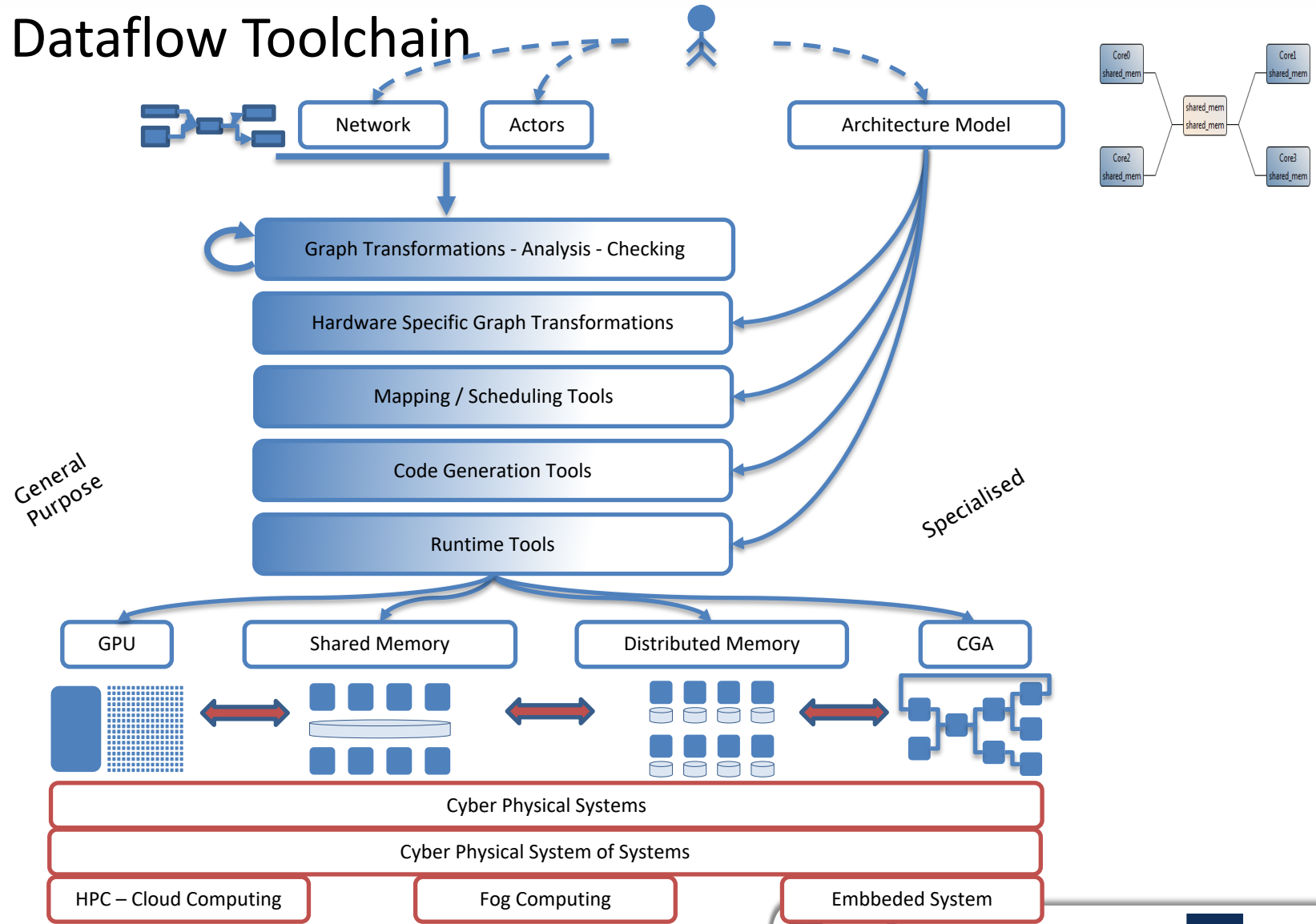


[LM87a] E. Lee and D. Messerschmitt, "Synchronous data flow", Proceedings of the IEEE, 1987.

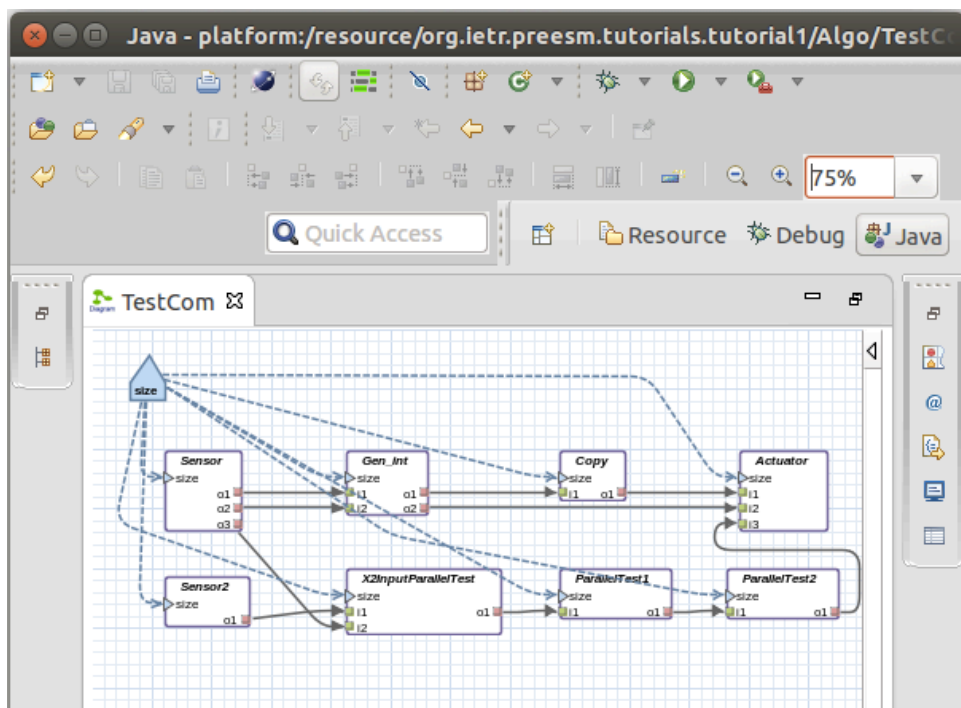
Interface-Based Synchronous Dataflow (IBSDF)



- Dataflow Toolchain



Open source project developed in collaboration with Texas Instruments, IBM, Kalray, etc.



Solution Gantt

Time: 0 cycles, 250 cycles, 500 cycles

Operators: Core0, Core1

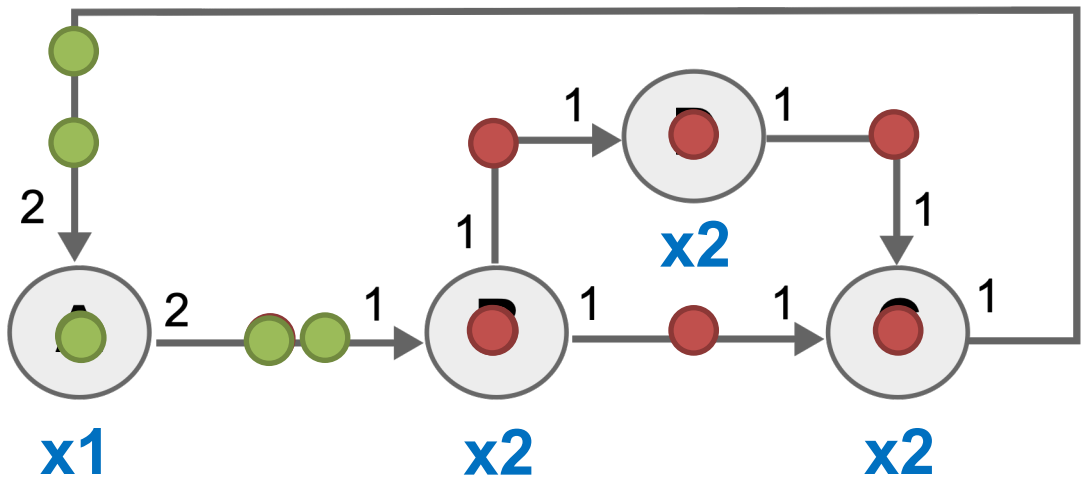
Tasks: Sens, Gen_i, Copy, X2Inp, Paral, Paral, Actua

DFTools Workflow console

```
10:22: 1 Starting allocation with BasicA
10:22: 1 BasicAllocator allocates 10mem.
10:22: 1 Workflow Step: Generate xtend q
10:22: 1 Workflow execution finished: /o
```

Synchronous Dataflow (SDF) Graph Analysis

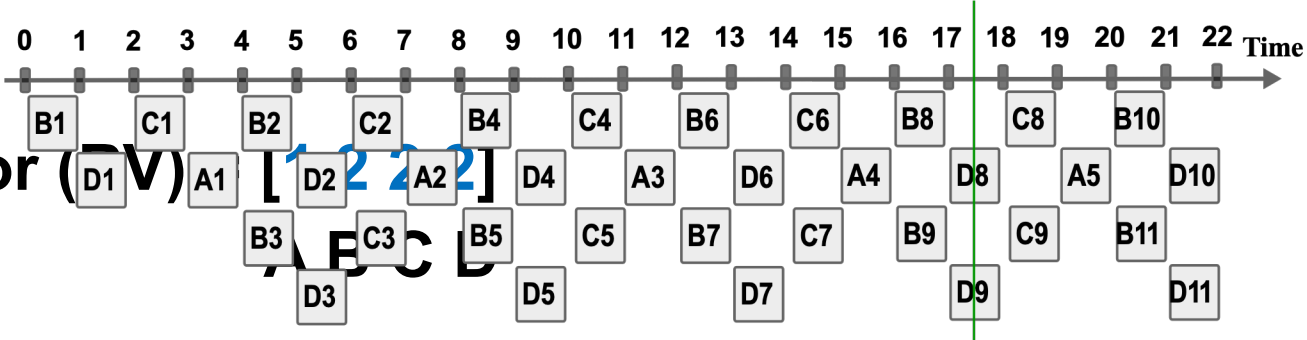
- Consistency
- Liveness
- Throughput
- Latency



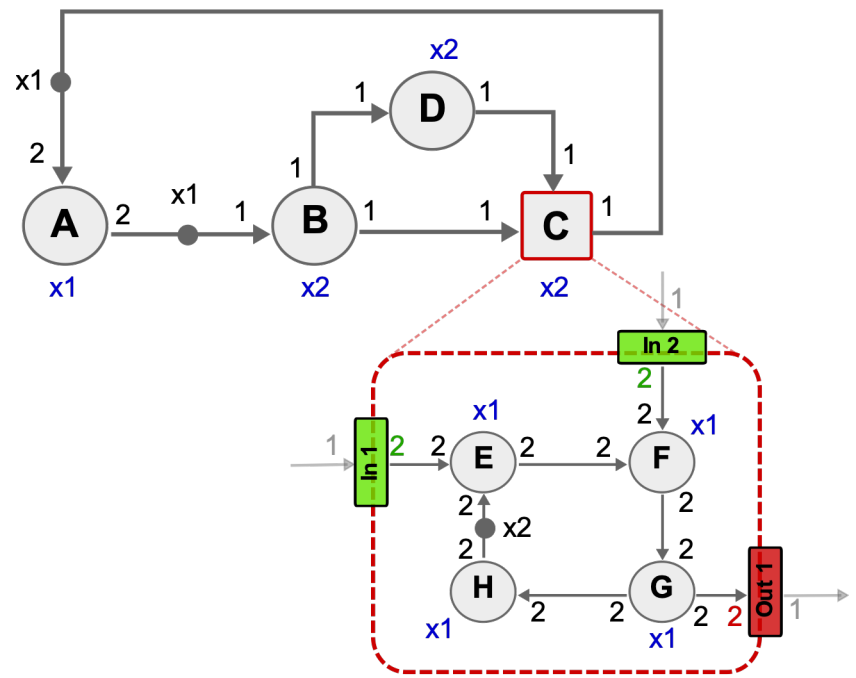
Graph analysis

Repetition Vector (RV)

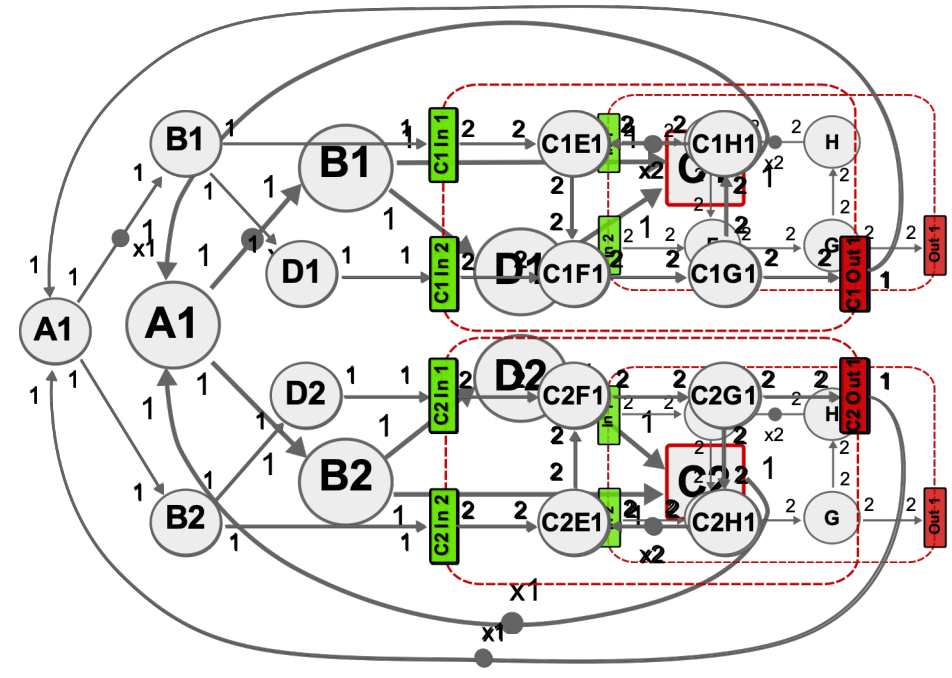
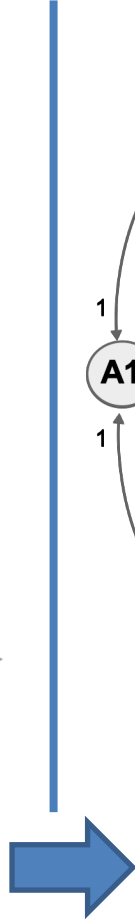
Simulation



Exemple of Graph Transformation



**IBSDF Graph
(Hierarchical)**



**Flatten Graph
(No Hierarchy)**

IBSDF Graphs		
Name	# Level	# Actor
Crypto	2	10
Large FFT (CSP)	2	10
LTE	4	18
Stereo Matching	2	41
Graph 1	3	15
Graph 2	5	20
Graph 3	6	24
Graph 4	5	150
Graph 5	8	240
Graph 6	10	100

Two IBSDF graph classes :

1. Real Applications
2. Generated Random Graphs



IBSDF Graphs		
Name	# Level	# Actor
Crypto	2	10
Large FFT (CSP)	2	10
LTE	4	18
Stereo Matching	2	41

Equivalent Flatten Graph	
# Acteur	# Arc
34	49
267	777
272	18
1606	3143

Graph 1	3	15
Graph 2	5	20
Graph 3	6	24
Graph 4	5	150
Graph 5	8	240
Graph 6	10	100

513	816
17 730	56 703
87 440	196 019
654 566	2 034 980
39 x10¹⁰	-
31 x10¹⁵	-

> 64 Gb of RAM

Development Process



Simpler
IBSDF Graph

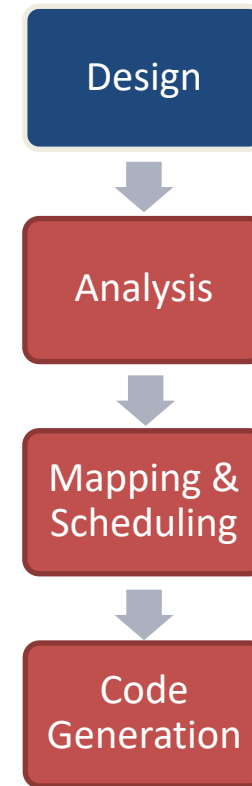


Complex
Flatten Graph

Conversion



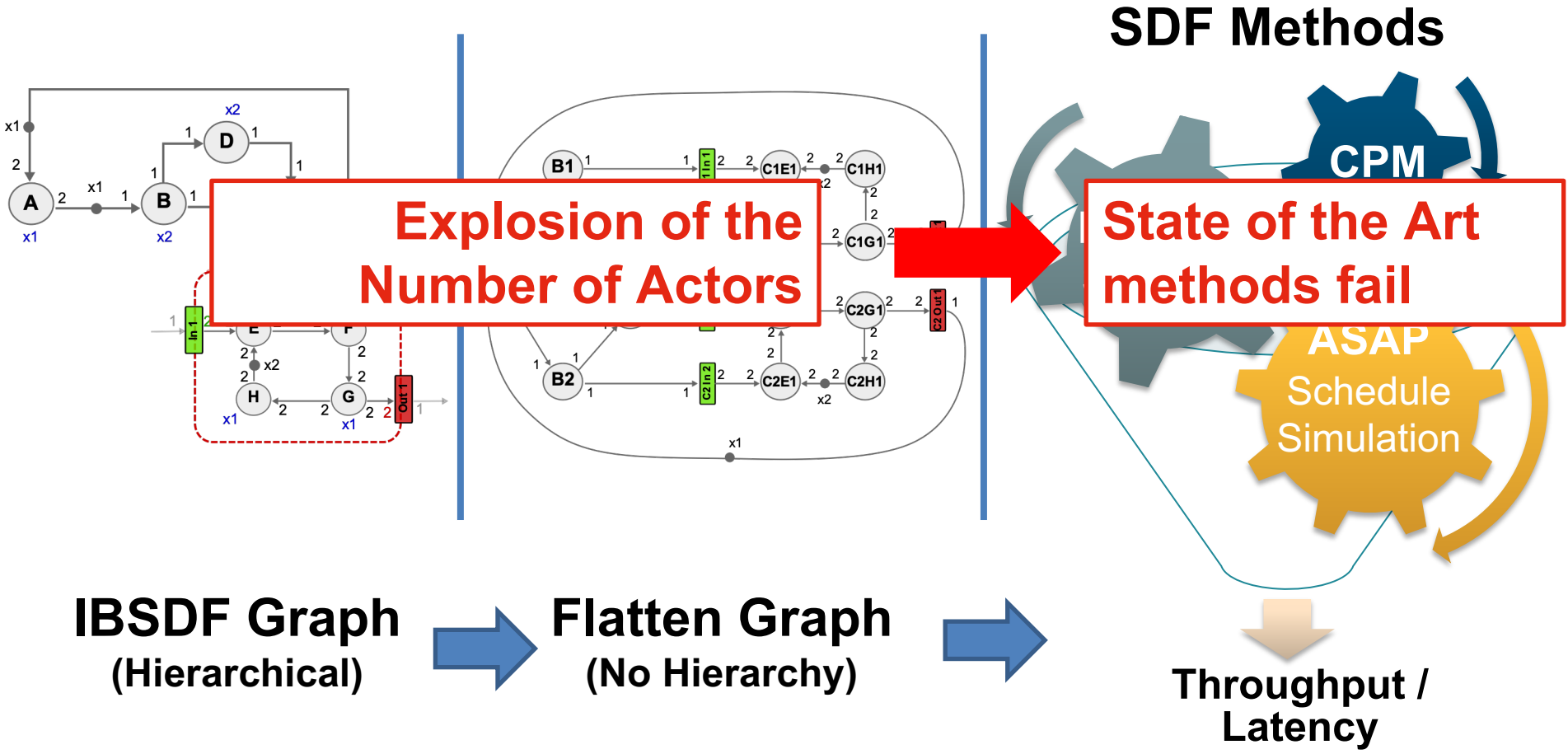
Latency / Throughput
evaluations



IBSDF Graphs			Flatten Graph	IBSDF Graph Transformation	
Name	# Level	# Actor	# Actor	SR	ESR
				# Actor	# Actor
Crypto	2	10	34	10	30
Large FFT (CSP)	2	10	267	4	8
LTE	4	18	272	3	7
Stereo Matching	2	41	1606	39	56
Graph 1	3	15	513	15	23
Graph 2	5	20	17 730	12	24
Graph 3	6	24	87 440	12	22
Graph 4	5	150	654 566	90	95
Graph 5	8	240	39 x10¹⁰	150	174
Graph 6	10	100	31 x10¹⁵	50	68

IBSDF Graphs		SDF Method		SR		ESR	
Name	# Level	Quality	Time	Quality	Time	Quality	Time
Crypto	2	100%	6 ms	100%	38 ms	100%	45 ms
Large FFT	2	100%	33 ms	100%	36 ms	100%	74 ms
LTE	4	100%	20 ms	100%	32 ms	100%	58 ms
Stereo Mat.	2	100%	80 ms	99%	37 ms	100%	130 ms
Graphe 1	3	100%	30 ms	34%	34 ms	100%	59 ms
Graphe 2	5	100%	1,2 s	55%	34 ms	100%	70 ms
Graphe 3	6	100%	2,3 s	21%	34 ms	100%	90 ms
Graphe 4	5	100%	1 min	12%	61 ms	100%	0,3 s
Graphe 5	8	-	-	?	61 ms	?	0,5 s
Graphe 6	10	-	-	?	72 ms	?	2 s

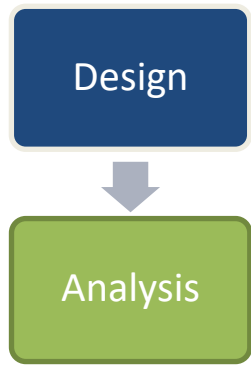
IBSDF Graphs		SDF Method	H-CPM	SR	
Name	# Level	Time	Time	Hier./Relaxed	Time
Crypto	2	0,65 ms	0,50 ms	1,00	0,50 ms
Large FFT	2	4,66 ms	3,73 ms	1,00	0,50 ms
LTE	4	2,52 ms	1,49 ms	1,00	0,50 ms
Stereo Mat.	2	18,20 ms	15,00 ms	1,01	0,87 ms
Graphe 1	3	2,40 ms	1,83 ms	1,80	0,93 ms
Graphe 2	5	108,16 ms	3,91 ms	1,76	0,99 ms
Graphe 3	6	441,40 ms	5,11 ms	3,73	1,00 ms
Graphe 4	5	6,25 s	48,14 ms	7,70	1,20 ms
Graphe 5	8	-	83,32 ms	5,50	4,74 ms
Graphe 6	10	-	308,00 ms	8 x10 ⁶	12,81 ms



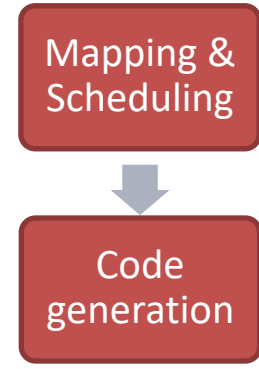
Development Process



Simplicity IBSDF Graph



Complexity Flatten Graph



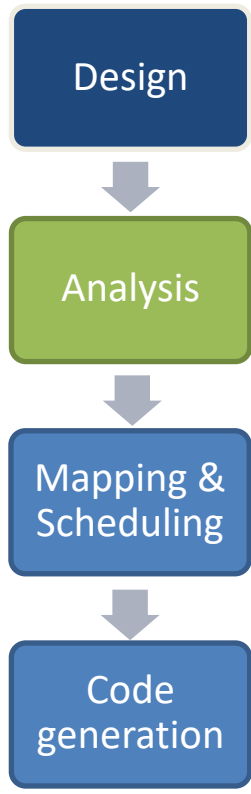
Conversion



Development Process



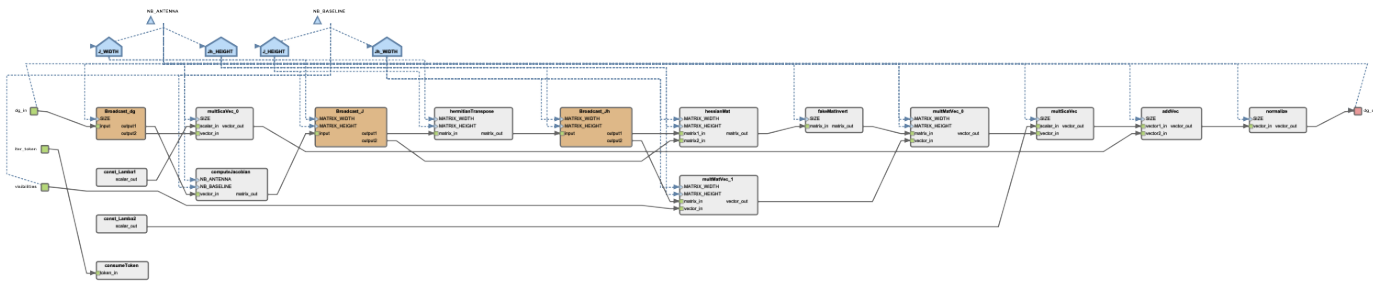
Simplicity IBSDF Graph



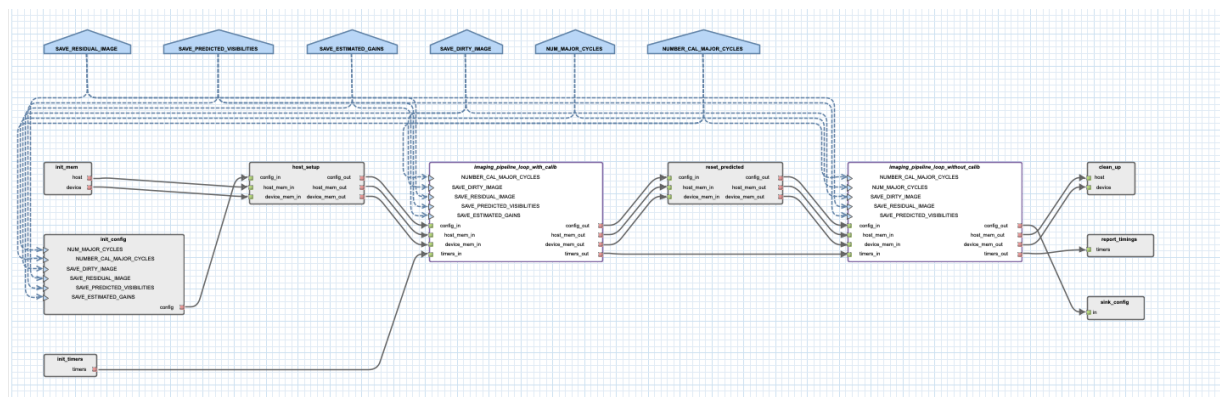
Complexity Flatten Graph



SDP Direction Independent Calibration



Auckland SDP Evolutionary Pipeline (SEP) Imaging Chain



DDFacet – KillMS Algorithms for direction dependant algorithms

$$\begin{aligned} & Nb_{acteurs} \\ &= NbMajorCycleCal \\ &* (Nb_{MinCycleCal} * 7 + NbCalCycle * 5 + 54) \\ &+ NbMajorCycleImg * (Nb_{MinorCycleImg} * 7 + 54) + 25 \end{aligned}$$

Data Level

Computation Level

Hardware Level

- Data size reduction
 - Spatial resolution
 - Time resolution
 - Precision Optimization
 - Standard / Custom types
 - Fixed / Floating points
 - Less up-to-date data (Relaxed Synchronization)
-
- Computation Skipping
 - Incremental Refinement
 - Loop Performation
 - Early Termination/Iterative Refinement
 - Block Skipping
 - Computation Approximation
 - Parameter Modification
 - Fuzzy Memorization
 - Mathematical Approximation
-
- Approximate Processing Units
 - Inexact Adders
 - Inexact Multipliers
 - Circuit Parameters Modification
 - DVFS
 - Clock Gating
 - Unreliable Storage

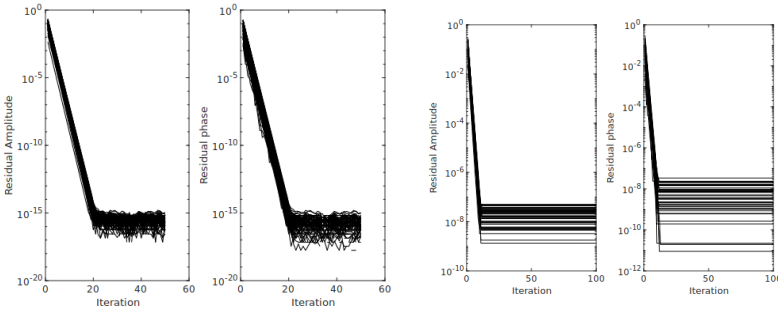
Data Level

- Data size reduction
 - Spatial resolution
 - Time resolution
- Precision Optimization
 - Standard / Custom types
 - Fixed / Floating points
- Less up-to-date data (Relaxed Synchronization)

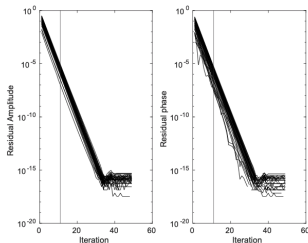
SDP Direction Independent Calibration

Floating Point 64

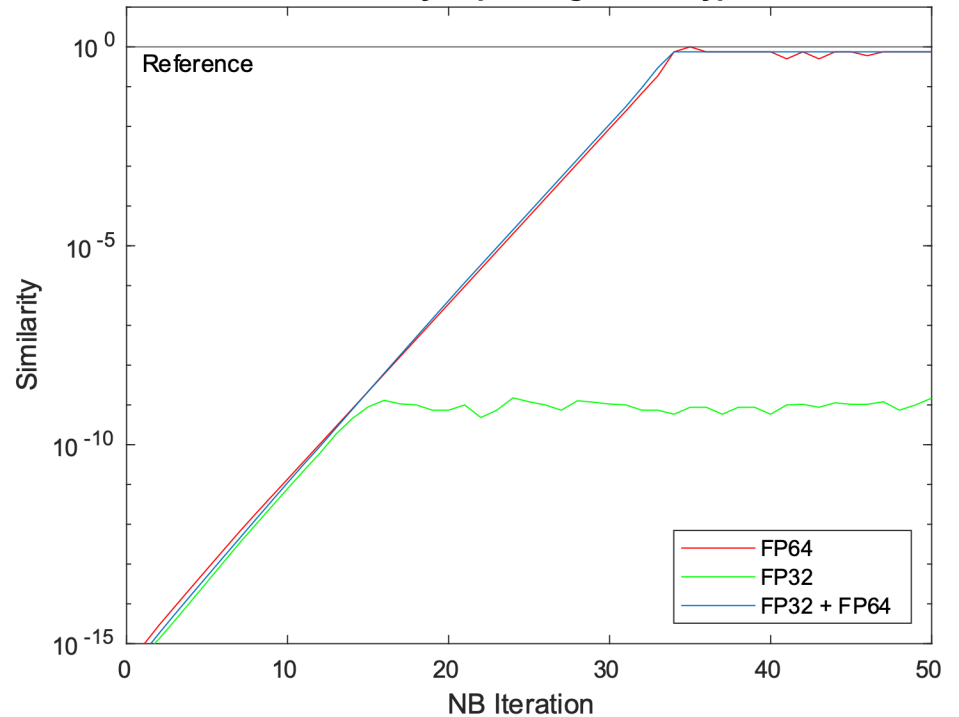
Floating Point 32



Mixed FP32 / FP64



Results accuracy depending on datatypes



- Simulation tools : Simgrid
- HPC Runtimes : Dask, StarPU
- Dataflow semantics : Daliuge

Thanks for Listening !