Scaling of 3D Game Engine Workloads on Modern Multi-GPU Systems

Jordi Roca Monfort (Universitat Politècnica Catalunya)
Mark Grossman (Microsoft)

Outline

- Introduction on Multi-GPU rendering
- RTT surface synchronization alternatives
- Multi-GPU performance models
- Scaling results
- Conclusion

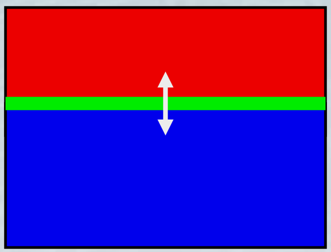
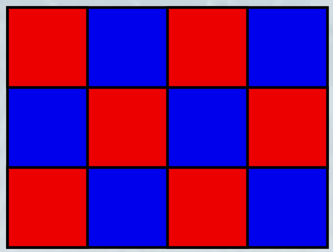
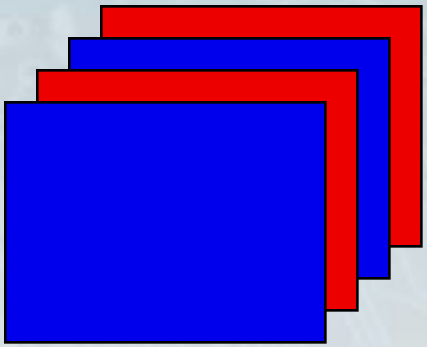
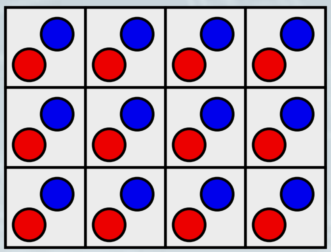
Multi-GPU rendering

- **Main purpose: Several GPUs collaborate to render frames of the same 3D scene.**
- Rendering task is massive parallel.
- High resolution scenes require a lot of pixel fillrate and memory BW.
- GPUs partial renders are composed to the final frame sequence.
- **Other different usages:**
- Multi-display rendering: each GPU renders a different viewport/screen.
- GPUs take different tasks: graphics rendering, physics and AI processing.

A choice for enthusiast gaming

- **Performance/scaling**:
- Usually high for few GPUs and high resolutions (1600x1200 minimum)
- Greatly depends on driver maturity and the game workload: Crysis Warhead hits 1.6x and Lost Planet hits 1.9x with 2-GPU systems.
- **Power**:
- Two graphics cards spend more than the equivalent single GPU solution targeting the same performance.
- **Upgrade cost:**
- Double performance for next game generation by acquiring a second graphics card (same GPU family and similar range counterpart).
- Extra cost of acquiring a high end motherboard with multiple PCIe ports.

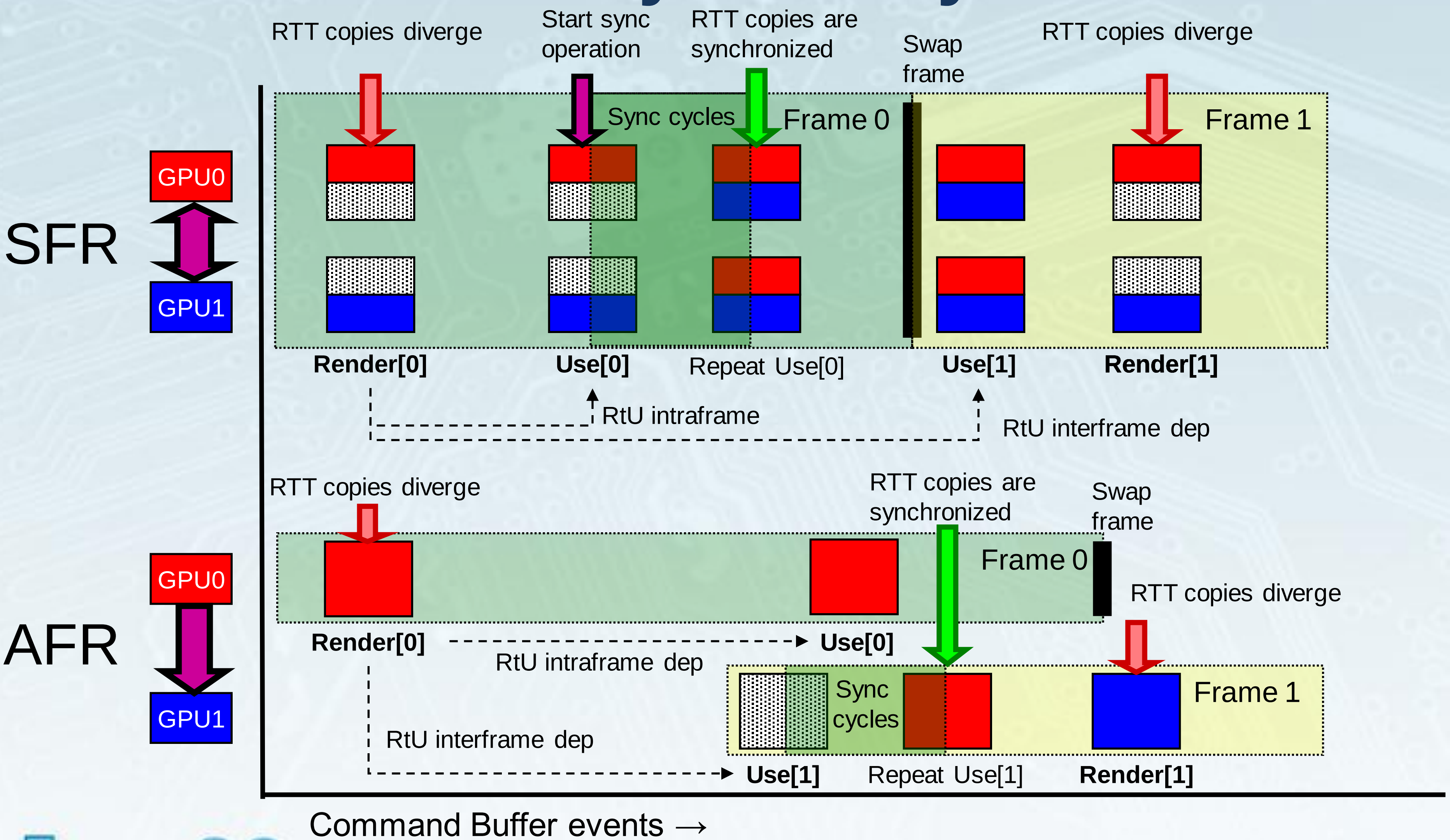
Rendering workload balance

Split Frame Rendering	Alternate Frame Rendering	SuperAA
 Dynamic split line  Fixed tiles (32x32)		
Geometry scaling problem	Decreased interactivity problem	Play at High AA modes (x16)

What do GPUs communicate to each other?

- Today's 3D engines don't just render the main backbuffer:
  - Draw commands can render to special surfaces used later as textures: reflections, shadow maps, lens flare, post-filtering ops,..
  - New draw dependency chain with **render-to-texture surfaces** as synchronization points.
- GPUs must **exchange** updated surface contents at this points to ensure data integrity → Inter-GPU syncs.

Render-to-Use sync analysis



Render-to-Use sync analysis

Game/Timedemo	Engine	Release date	Screen resolution	Frames	RTT surfaces	% RTT time
3DMark06/Canyon Flight	Proprietary	2006/01	16 x 12 (1AA)	3344	23	69.11%
FEAR/PerformanceTest	LithTech	2005/10	16 x 12 (1AA)	4130	5	4.48%
Call Of Duty 2/carentan	Proprietary	2005/10	16 x 12 (1AA)	1355	8	23.47%
Call Of Duty 2/demo5	Proprietary	2005/10	16 x 12 (1AA)	1210	8	18.87%
Company Of Heroes/Intro	Essence	2006/09	16 x 12 (1AA)	12195	5	85.74%
Half Life 2 Lost Coast/VST	Source	2005/10	25 x 16 (8AA)	9712	5	15.88%
BattleField 2142/suez canal	Refractor2	2006/10	25 x 16 (1AA)	7400	8	80.05%
BattleField 2/abl-chini	Refractor2	2005/06	16 x 12 (1AA)	8217	3	4.15%

Render-to-Use sync analysis

RTT Surface Id	Intraframe (SFR)		Interframe (AFR)	
	RtU deps	syncs	RtU deps	syncs 2-GPU
306	335442	7405	0	0
307	35722	28322	0	0
308	14805	14805	0	0
309	14811	14811	0	0
30a	2	1	0	0
30b	1	1	0	0
30d	14516	4713	4007	9
30e	252642	7305	0	0

Game/Timedemo						
3DMark06/Canyon Flight						
FEAR/PerformanceTest						
Call Of Duty 2/carentan						
Call Of Duty 2/demo5						
Company Of Heroes/Intr						
Half Life 2 Lost Coast/VS						
BattleField 2142/suez canal	Refractor2	2006/10	25 x 16 (1AA)	7400	8	80.05%
BattleField 2/abl-chini	Refractor2	2005/06	16 x 12 (1AA)	8217	3	4.15%

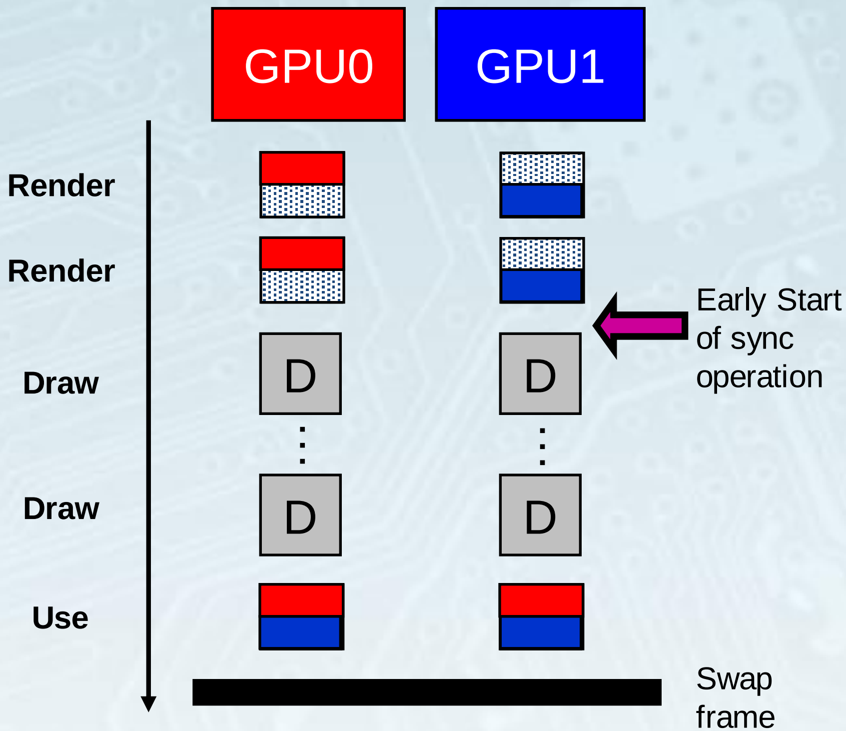
- Track per-surface RtU dependencies and required syncs.
- The more intraframe syncs, the worse SFR performs.
- The more interframe syncs, the worse AFR performs.

The contributions of this work

- **Which characteristics make 3D Games suitable for multi-GPU systems?**
- Render-to-texture sync analysis.
- Do they enable any optimization? (See next section)
- **Can we measure multi-GPU scaling based on 3D game workload characteristics?**
- Using a simplified model.
- Using real 3D workload data.
- Evaluate SFR, AFR and combined modes (4+ GPUs).

RTT surface synchronization alternatives

Leverage RtU gap: Early Copy



Game/ Timedemo	syncs	RtU sync Gap	% Pixel shading bound
3DM06	123691	0.42%	52.85%
FEAR	2426	83.68%	65.08%
COD2c	83780	1.70%	96.28%
COD2d	48175	1.65%	94.63%
COH	36626	0.01%	96.80%
HL2	19788	36.45%	21.95%
BF2142	77363	3.13%	87.16%
BF2	22553	30.40%	44.44%

RtU gap wrt frame duration

% of time the RTT draw is pixel shading bound

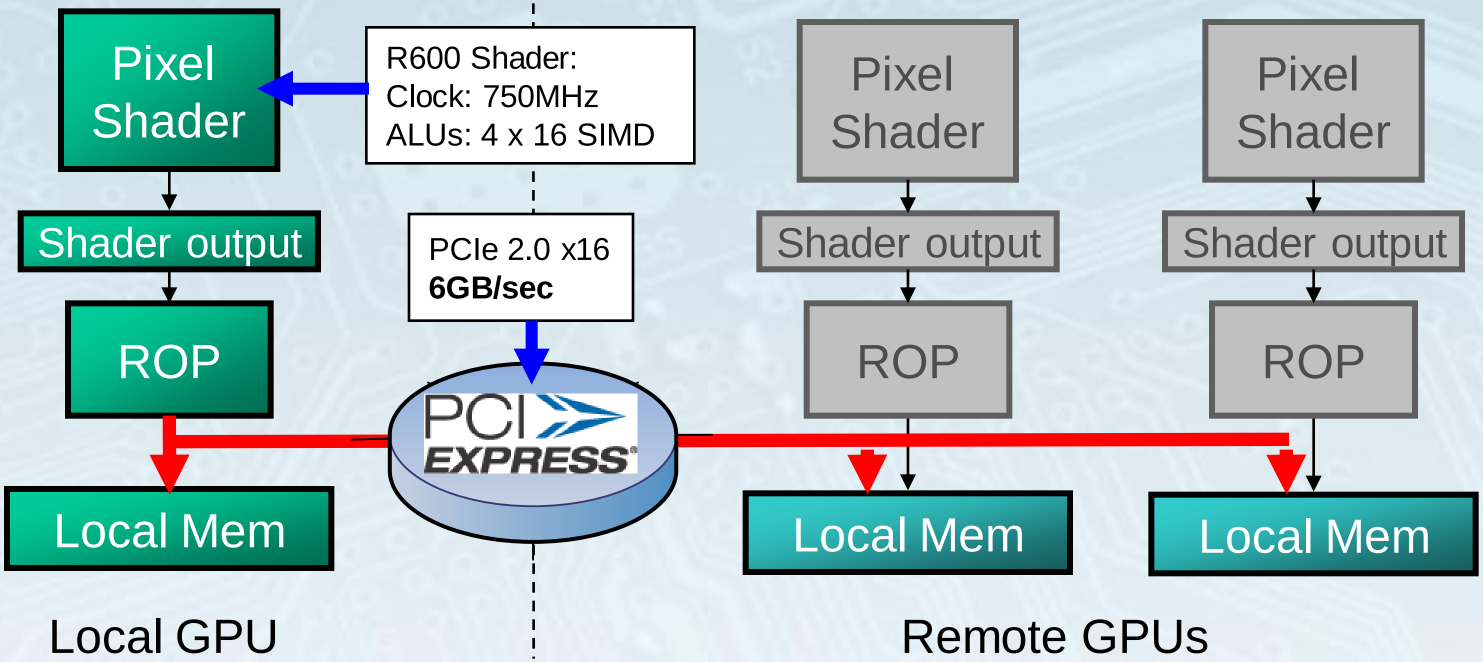
Determine last render:

- Use Prediction table,
- Look ahead command buffer.

Penalization cost =

Sync cycles (delayed copy) – RtU gap cycles

Pixel Shading bounds: Concurrent Update



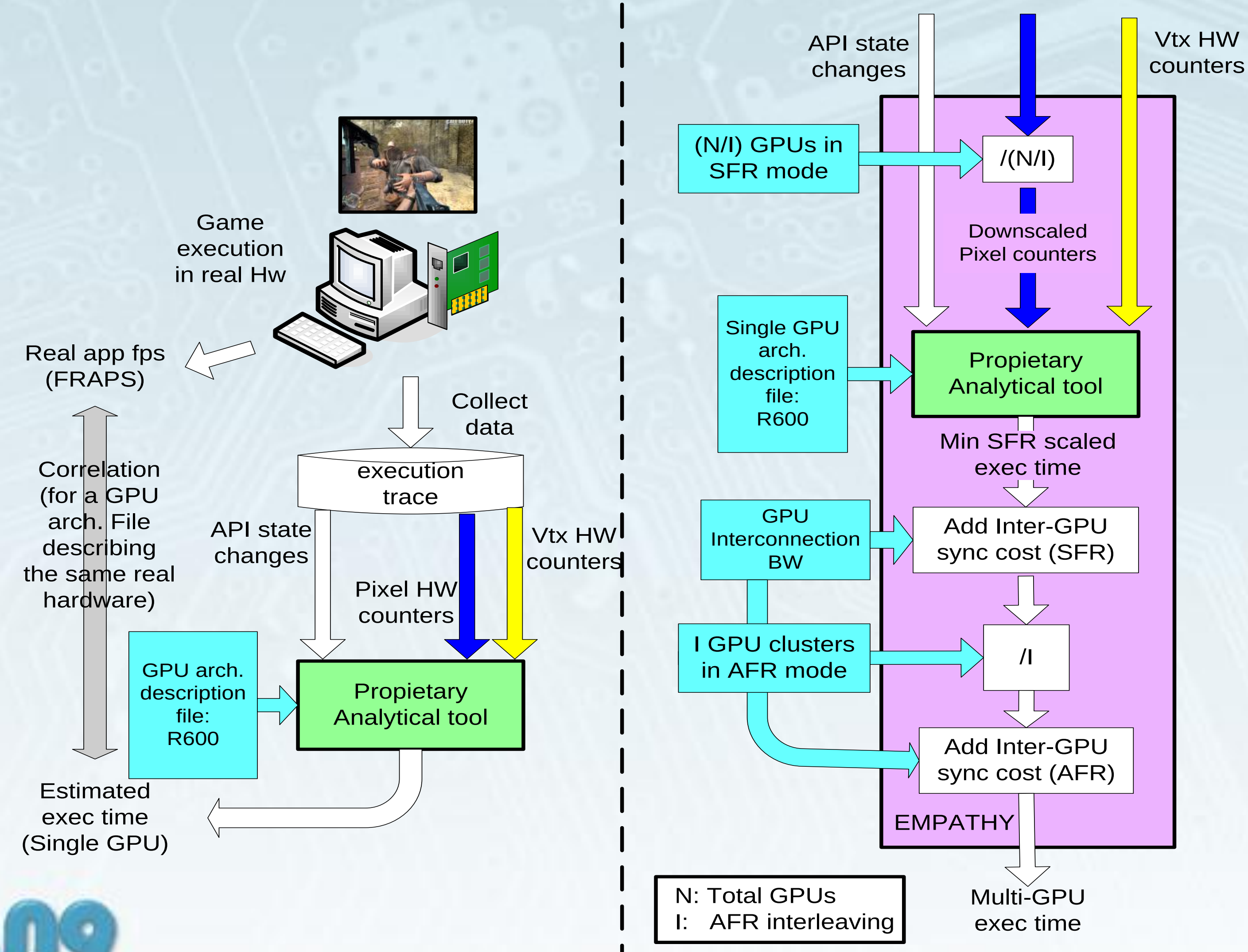
- Shading a thousand fragments (35 instructions) in the R600 SPUs takes 1000 x 35 / 64 = 547 cycles.
- Sending the corresponding color outputs through the PCIe 2.0 bus takes 1000 x 4 bytes x 750 MHz / 6 GB/s = 500 cycles

Penalization cost =

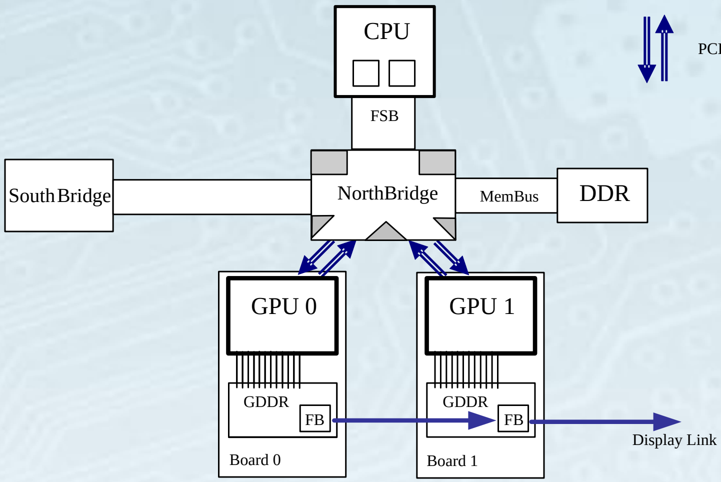
Remote sent cycles – Pixel shading cycles

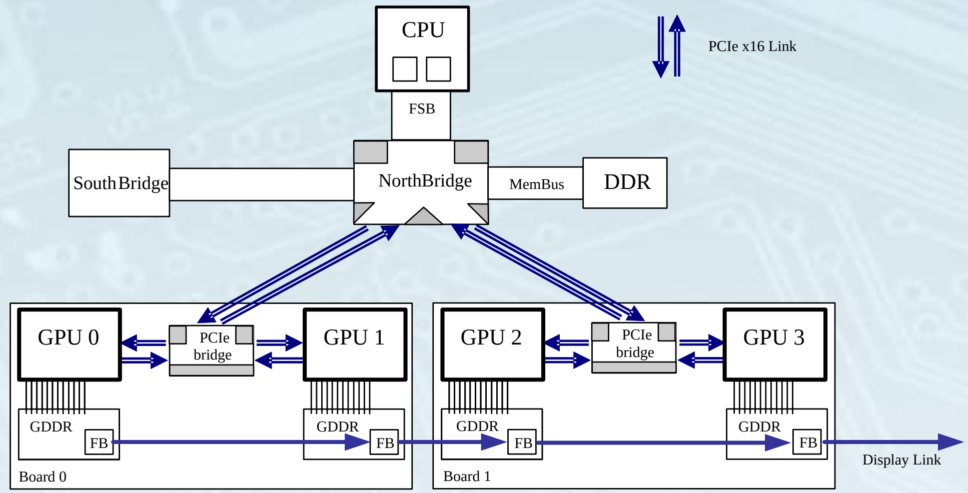
Multi-GPU Performance Models

EMPATHY analysis tool



Multi-GPU interconnection network



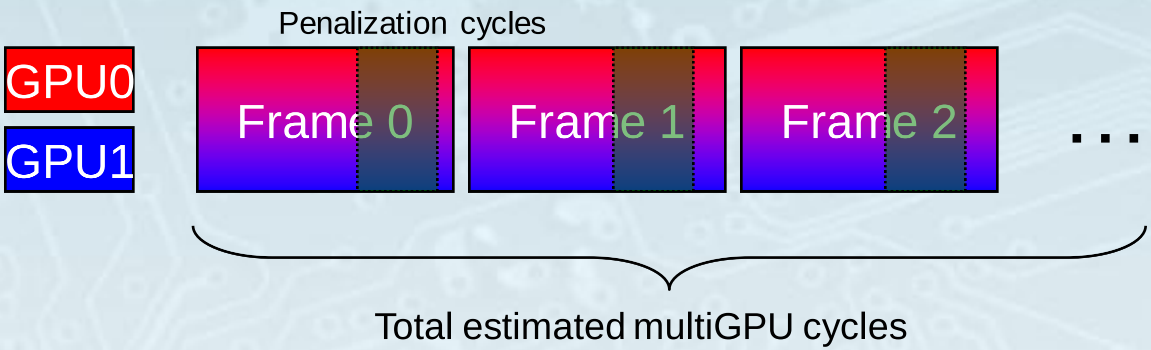


- SFR sync: all-to-all GPU simulatenous transfers.
- The NorthBridge PCIe ports become the bottleneck.
- Each GPU “sees” reduced peak BW as function of the number of GPUs:

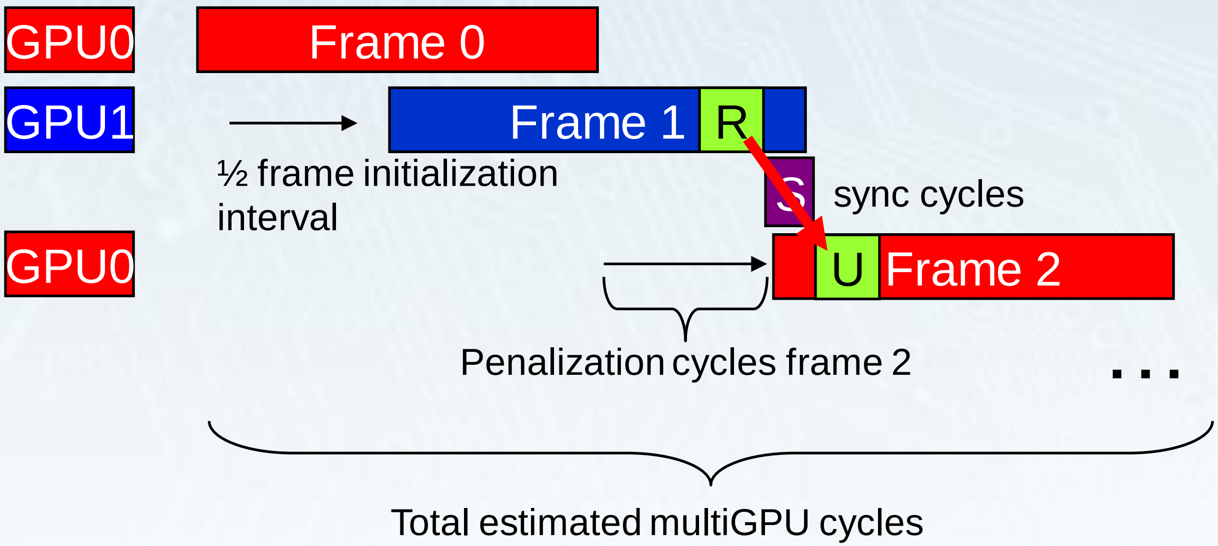
$$BW_{eff} = BW_{peak} * \left(\frac{N * (N-1) - N}{2} \right)^{-1}$$

Performance scaling models

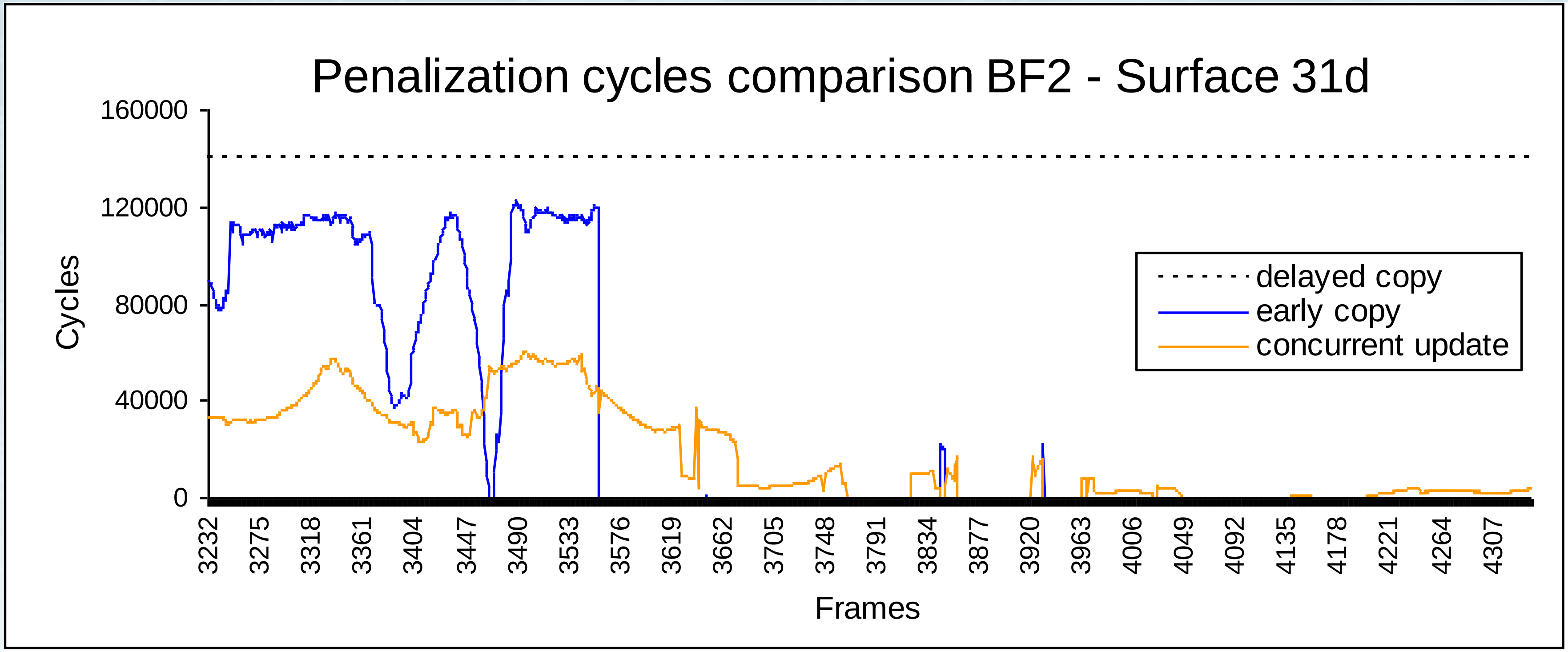
SFR



AFR



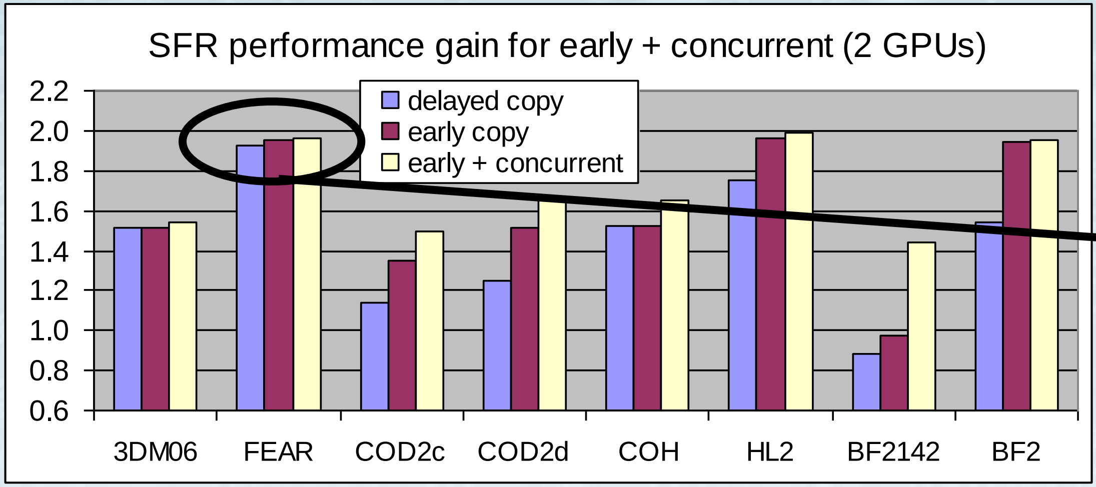
Early copy + concurrent cost (SFR)

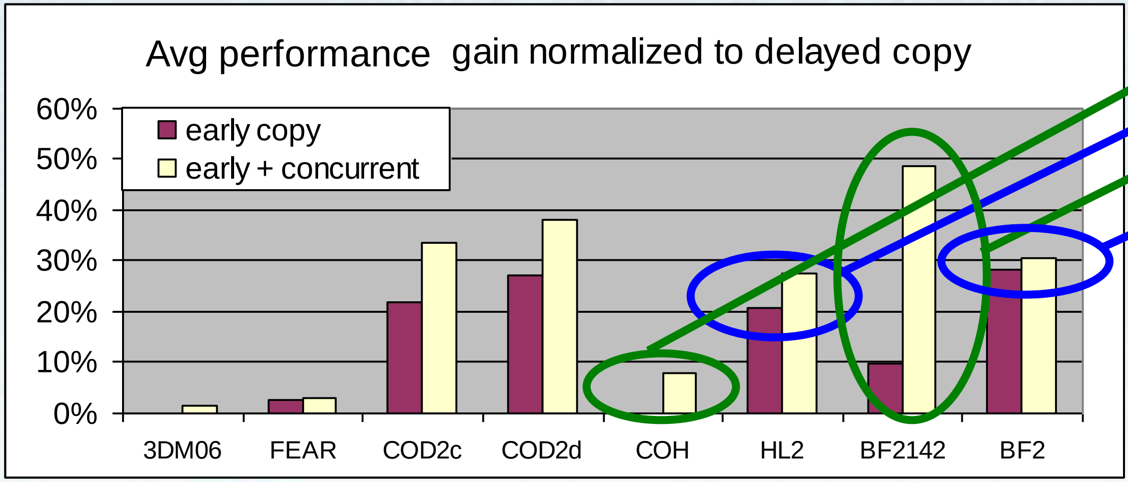


- Choose per surface the best sync alternative each frame that incurs in the minimum penalization cycles.

Scaling Results

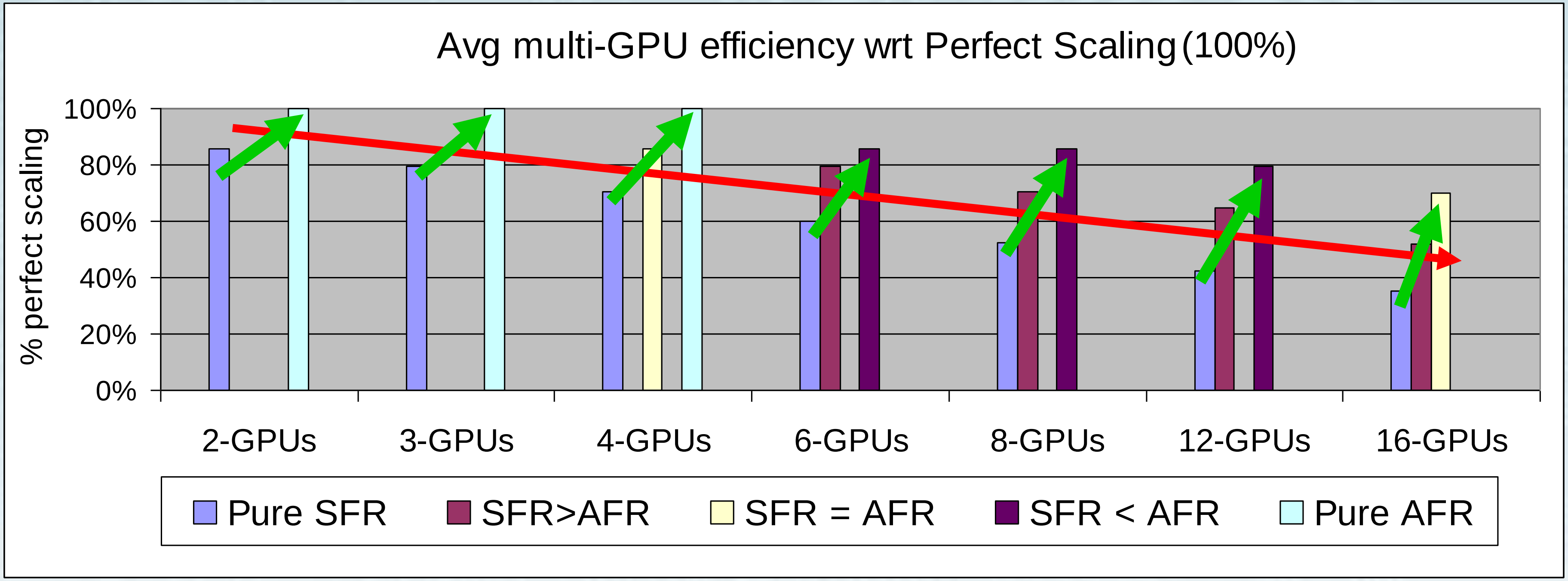
SFR scaling (early copy + concurrent update)





Game	syncs	RtU sync Gap	% Pixel shading bound
3DM06	123691	0.42%	52.85%
FEAR	2426	83.68%	65.08%
COD2c	83780	1.70%	96.28%
COD2d	48175	1.65%	94.63%
COH	36626	0.01%	96.80%
HL2	19788	36.45%	21.95%
BF2142	77368	3.13%	87.16%
BF2	22553	30.40%	44.44%

Combined SFR/AFR modes scaling results



- SFR scaling was tested using **Early Copy + Concurrent Update** optimization.
- Low interframe syncs benefits mostly AFR configurations.

Game	Interframe Syncs i2	Interframe Syncs i4
3DM06	0	0
FEAR	0	0
COD2c	0	0
COD2d	0	0

Game	Interframe Syncs i2	Interframe Syncs i4
COH	41	83
HL2	1442	1945
BF2142	9	26
BF2	1	3

Conclusion

Conclusion

- **Inter-GPU synchronization requirements of render-to-texture surfaces in 3D games impact multi-GPU performance/scaling.**
- **This work has evaluated the potential benefits of two proposed sync alternatives based on RTT update anticipation for a set of popular DX9 games.**
- Leverage of the RtU gap and the pixel shading cost increases SFR scaling.
- **This work has shown a simple multi-GPU performance analytic model based on real 3D game execution data, that allows to evaluate SFR, AFR and combined rendering modes.**
- Observed low interframe syncs benefit mostly AFR configurations.

Thank you!