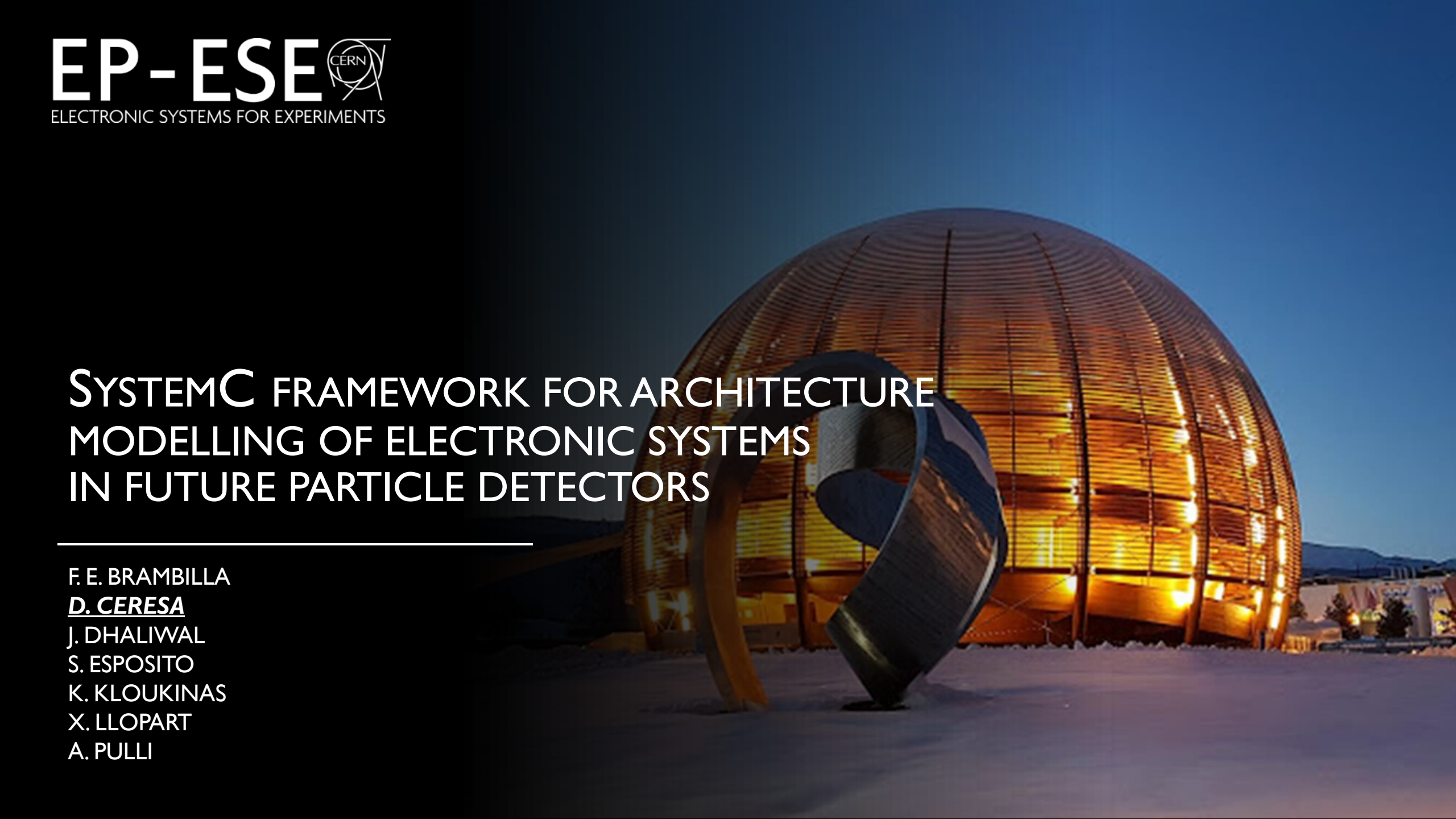
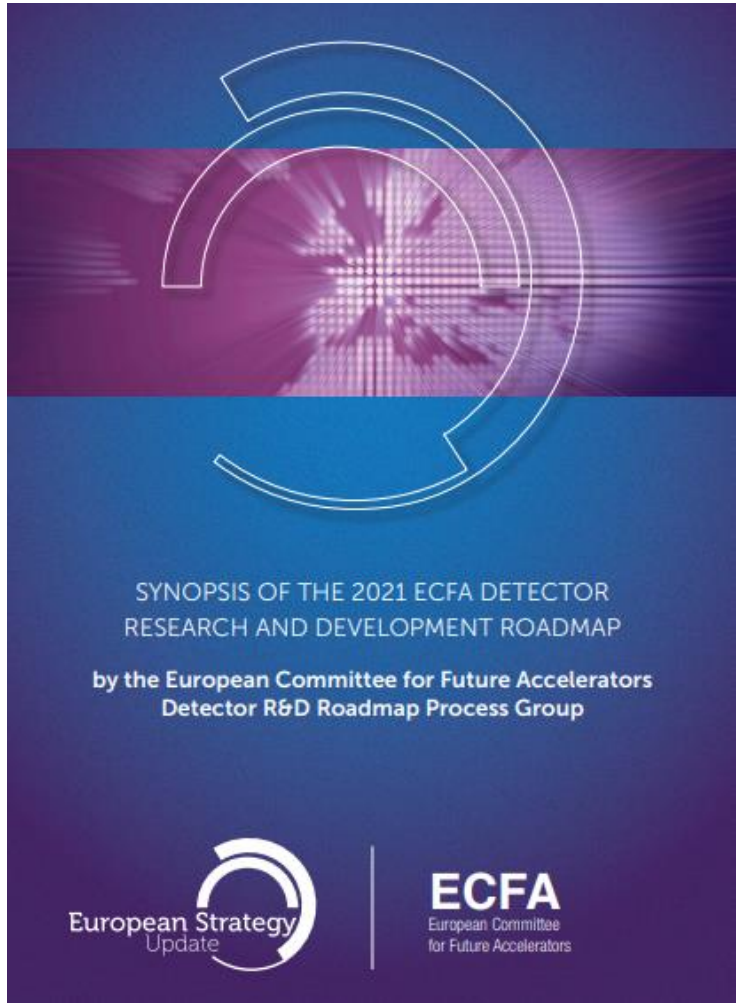
EP-ESE
ELECTRONIC SYSTEMS FOR EXPERIMENTS



SystemC framework for architecture modelling of electronic systems in future particle detectors

F. E. BRAMBILLA
D. CERESA
J. DHALIWAL
S. ESPOSITO
K. KLOUKINAS
X. LLOPART
A. PULLI

CONTEXT



DETECTOR RESEARCH AND DEVELOPMENT THEMES (DRDTs) & DETECTOR COMMUNITY THEMES (DCTs)

		< 2030	2030-2035	2035-2040	2040-2045	> 2045
Electronics	**DRDT 7.1** Advance technologies to deal with greatly increased data density					
	DRDT 7.2 Develop technologies for increased intelligence on the detector					
	DRDT 7.3 Develop technologies in support of 4D- and 5D-techniques					
	DRDT 7.4 Develop novel technologies to cope with extreme environments and required longevity					
	DRDT 7.5 Evaluate and adapt to emerging electronics and data processing technologies					

EP-R&D

Programme on Technologies for Future Experiments

SUMMARY | PROPOSAL 2020 - 2024 | CONTINUATION PROPOSAL 2024-2028

EP-ESE
ELECTRONIC SYSTEMS FOR EXPERIMENTS

IC TECHNOLOGY WORK PACKAGE (WP5)

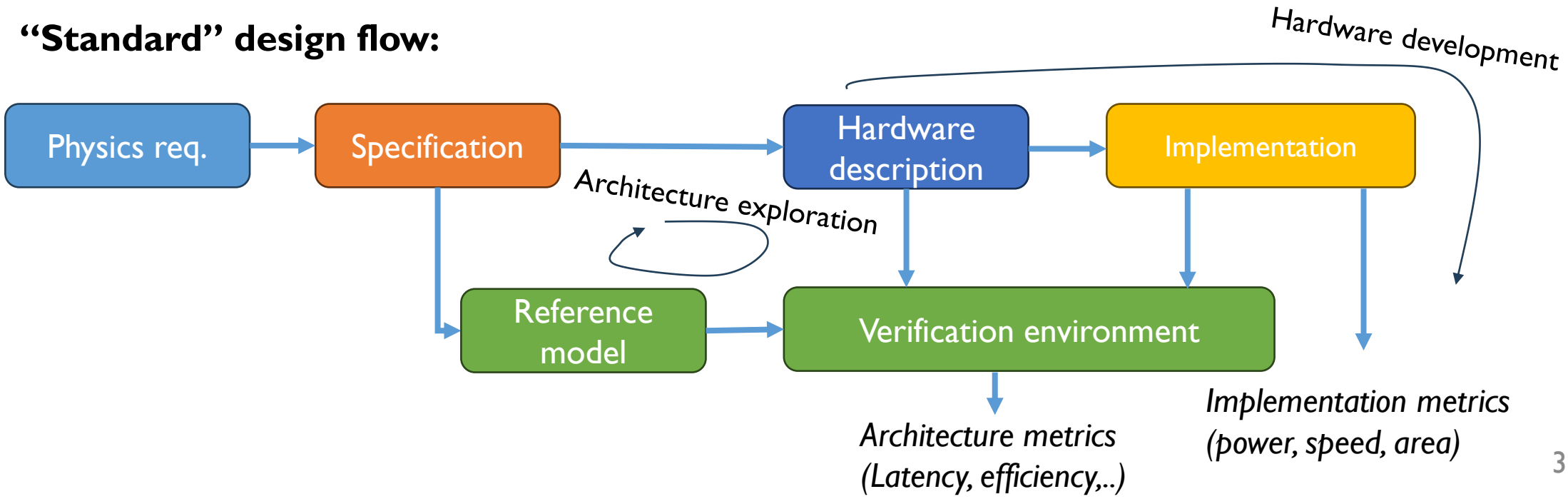
*MISSION: DEVELOP **INTELLIGENCE ON DETECTOR** SOLUTION*

ELECTRONIC SYSTEM DESIGN FLOW IN HEP

Limitation of the currently used design flow:

- Based only on a low-abstraction level description of the system (hardware level).
- Architecture exploration is time and resources-heavy
- in multi-chip modules/detector, single chip are optimized separately

"Standard" design flow:



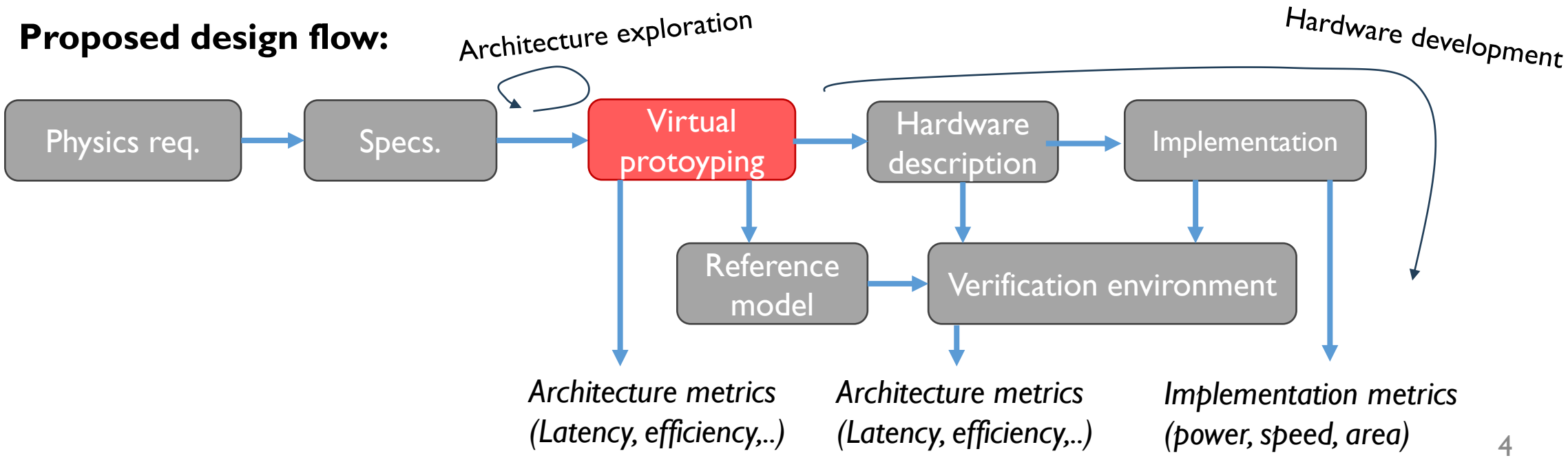
Electronic System Level approach

Develop a **high abstraction level** description of the system, from front-end to back-end, for:

- architecture exploration
- new feature development
- reference model development

Requires a self-contained environment for Virtual prototyping

Proposed design flow:



PixESL: an Electronic System Level prototyping framework

Open source:

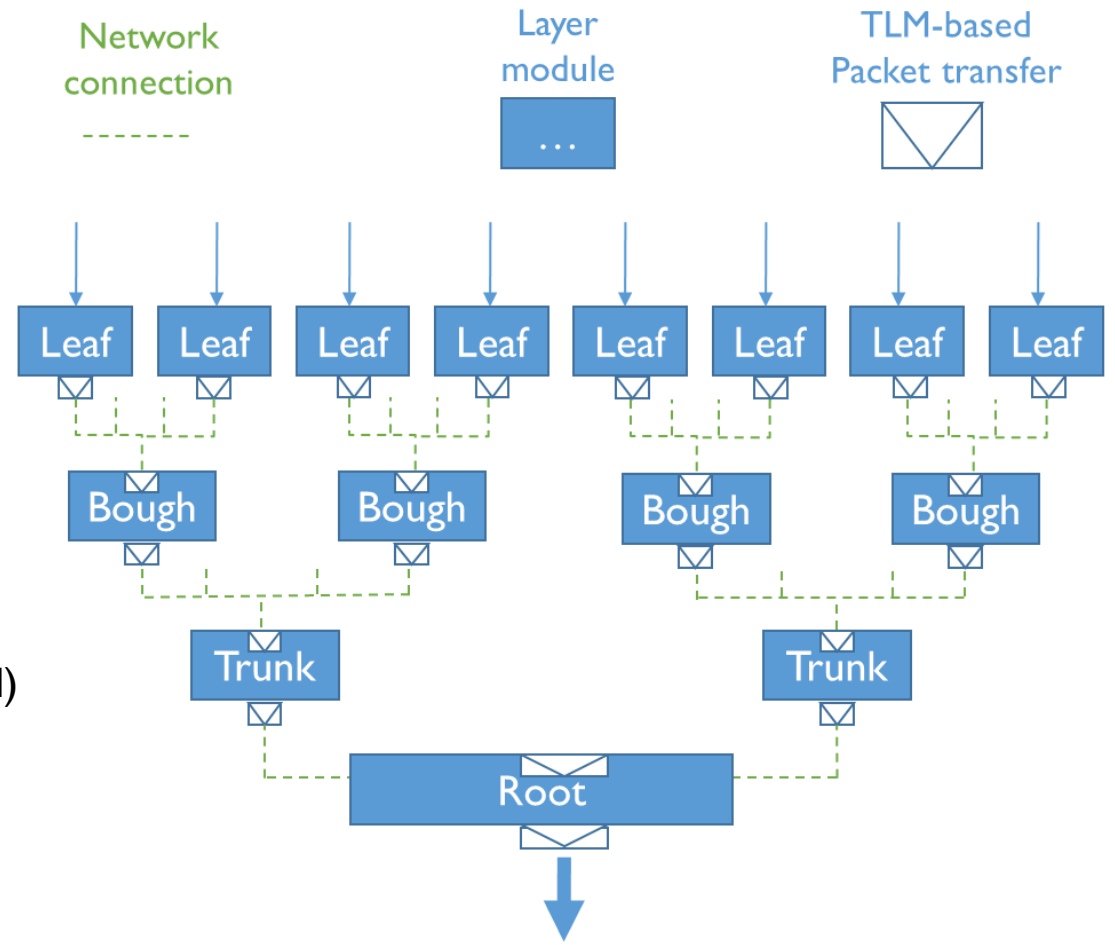
- The model is based on C++ and SystemC
- Performance analysis are based on Python

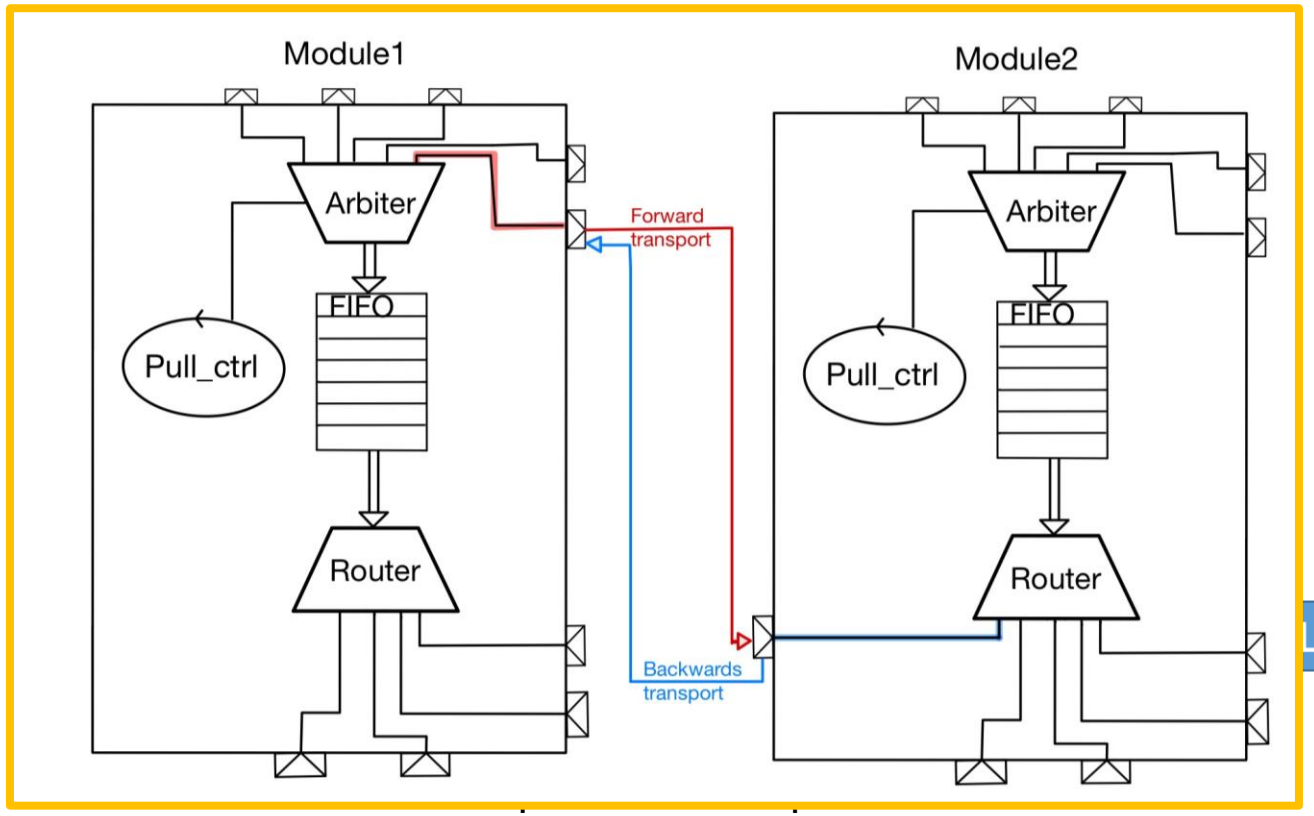
User-friendly:

- User and developer roles are separated
- The framework supports architectural and network configurability (structure, memory, arbitration, interconnections)

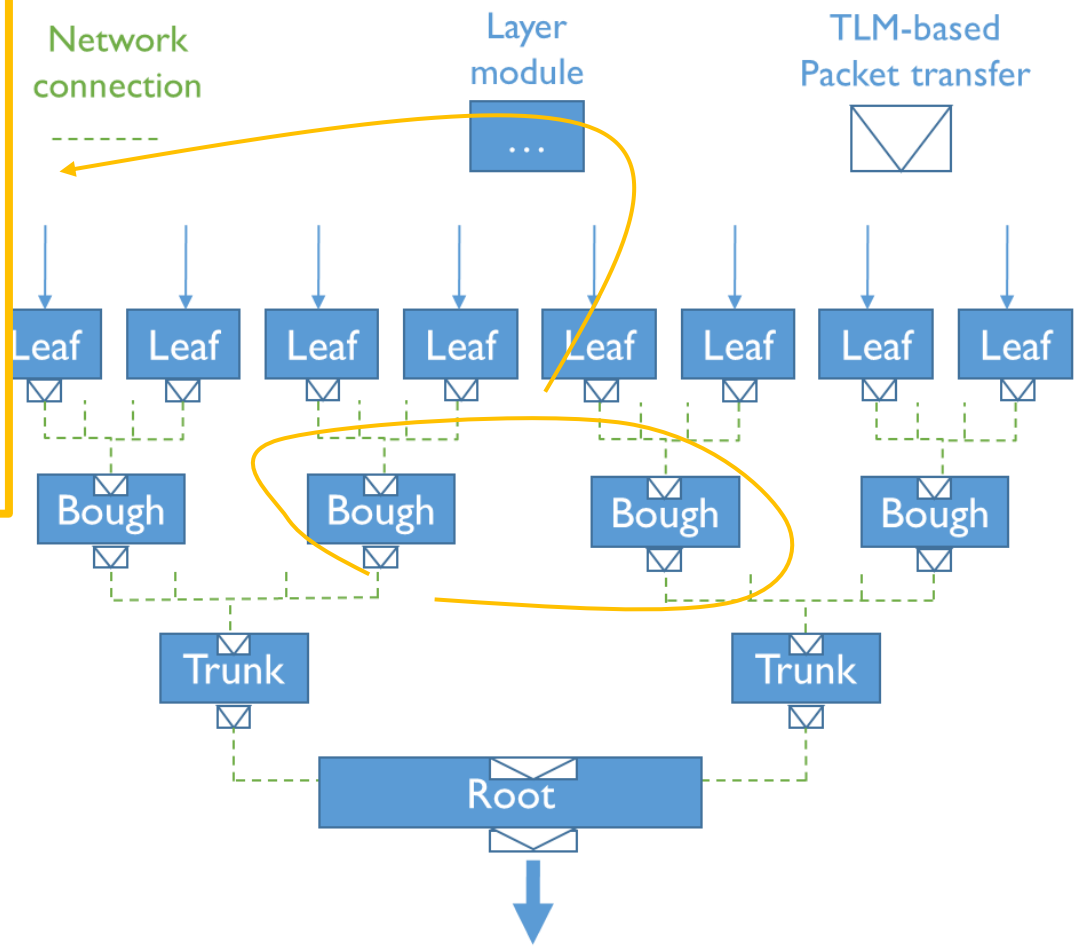
Reusable:

- Generalized layers and standardized packet transport (TLM)
- A library of layer types, functional components, and packet transport types
- Common integrated metrics analyzer





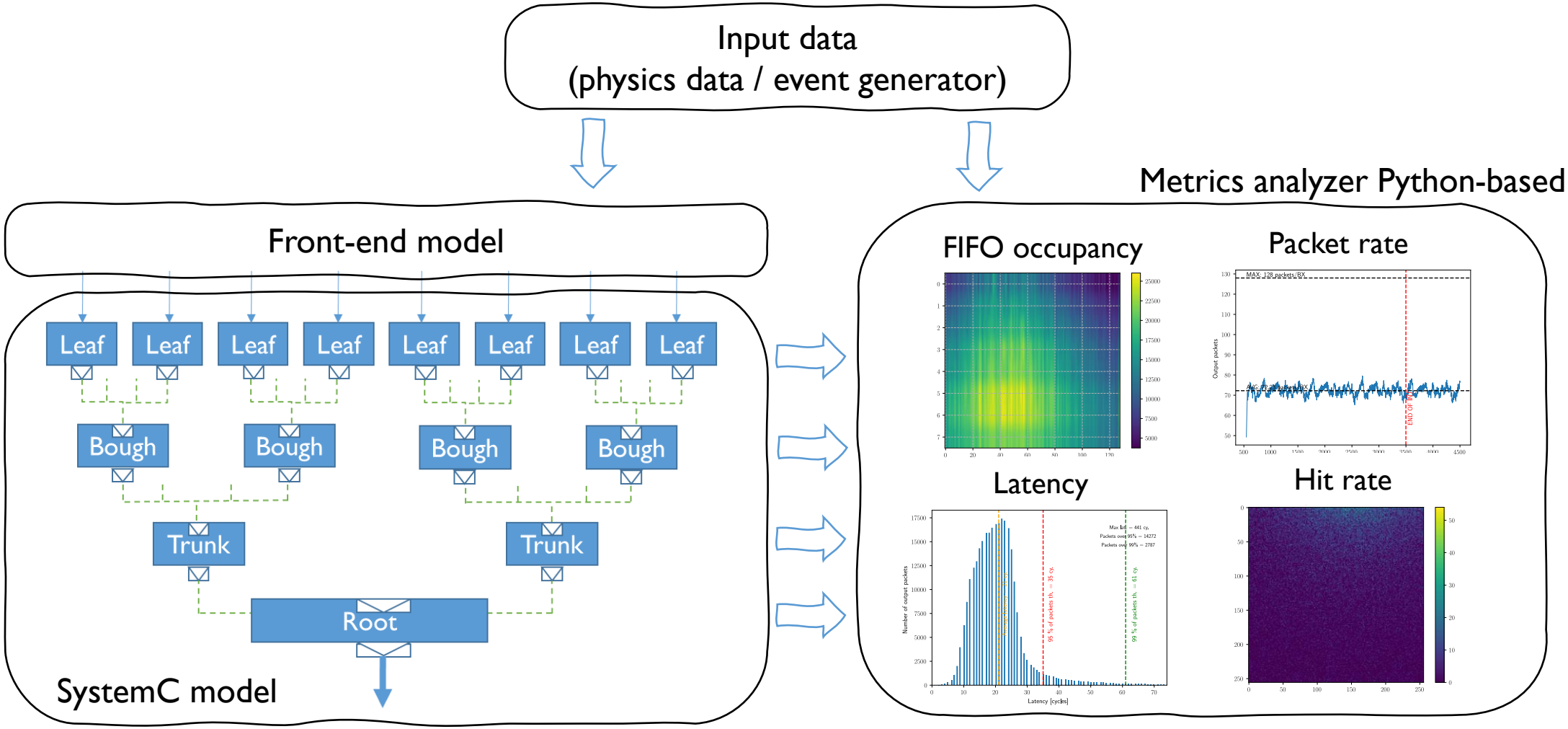
PROTOTYPING FRAMEWORK



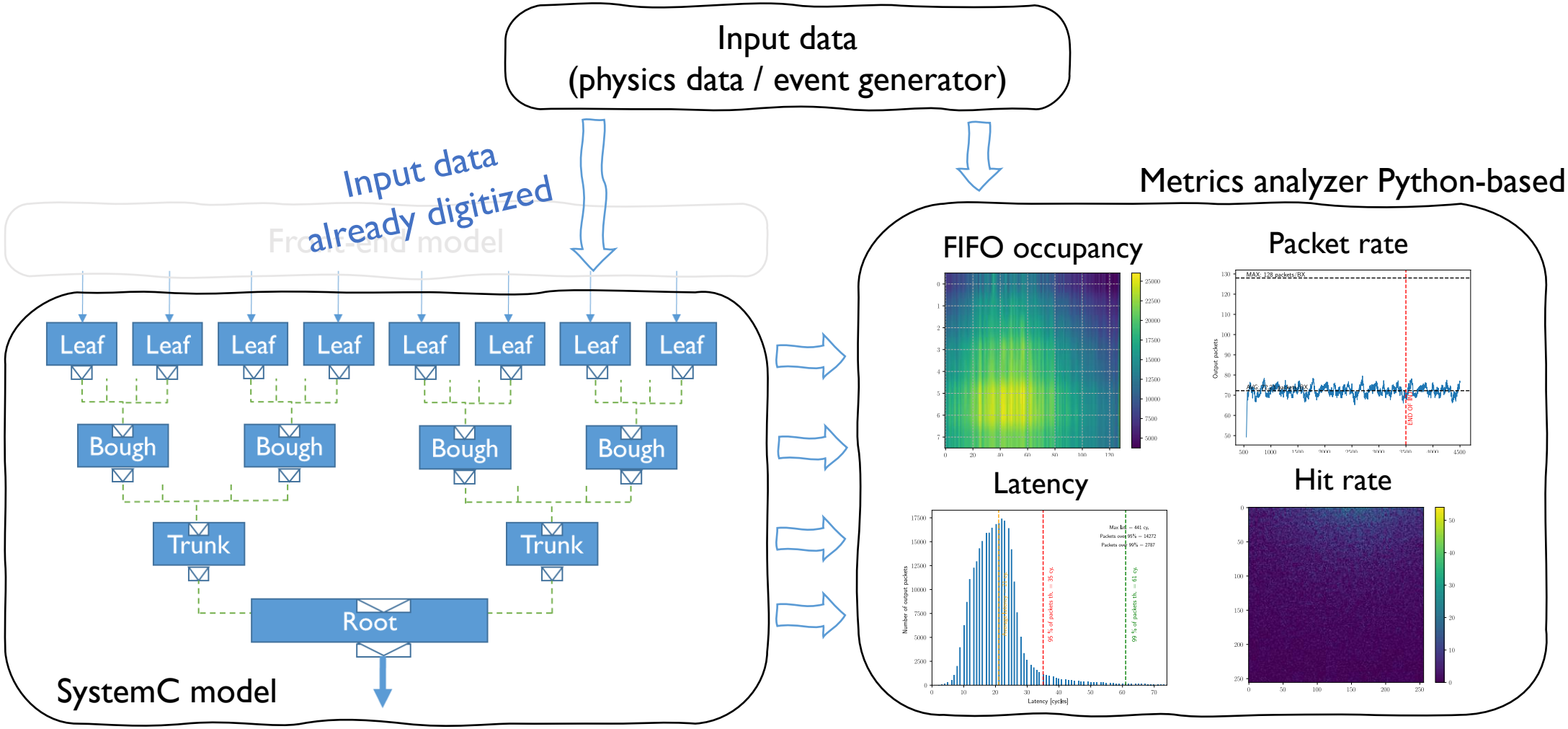
Reusable:

- Generalized layers and standardized packet transport (TLM)
- A library of layer types, functional components, and packet transport types
- Common integrated metrics analyzer

PIXESL: AN ELECTRONIC SYSTEM LEVEL PROTOTYPING FRAMEWORK



PIXESL: AN ELECTRONIC SYSTEM LEVEL PROTOTYPING FRAMEWORK

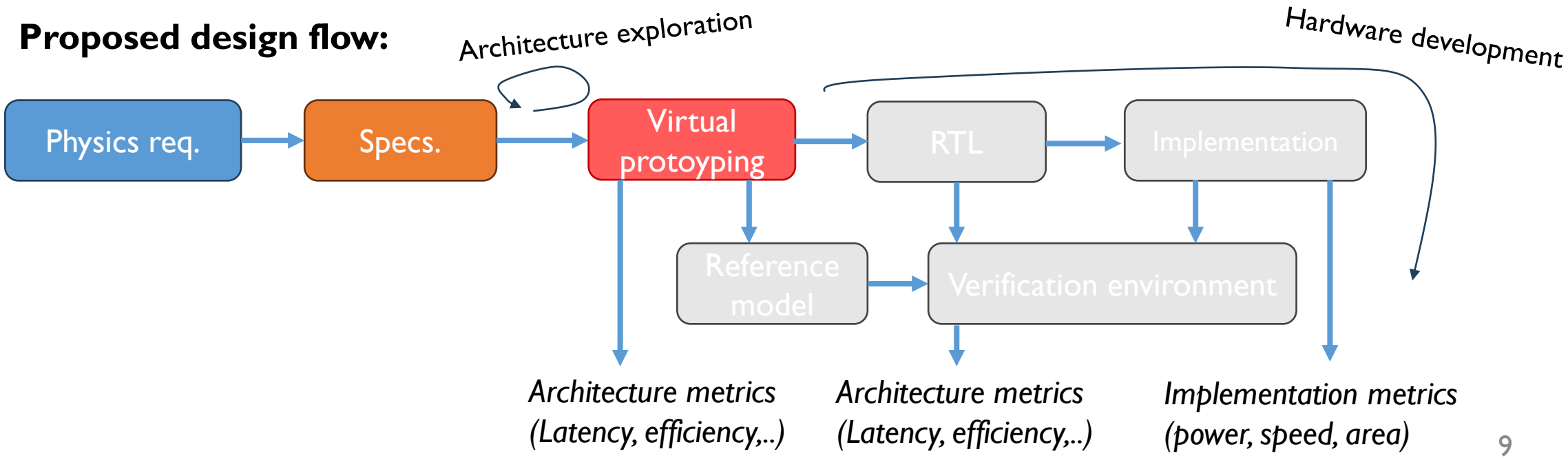


LHCb Velo upgrade II architecture exploration

Use the virtual prototyping framework to explore the Velo upgrade architecture:

- Run with real physics event
- Performed by a junior fellow and a master student with no experience in pixel ROC design
- Velopix model development ~ 1 month (first study case of PixESL)

Proposed design flow:



LHCb VELO upgrade II architecture exploration

The upgrade aims at a 4D pixel detector.

https://cds.cern.ch/record/2844669/

Main readout challenge:

extreme occupancy (x2 Velopix)

Flow:

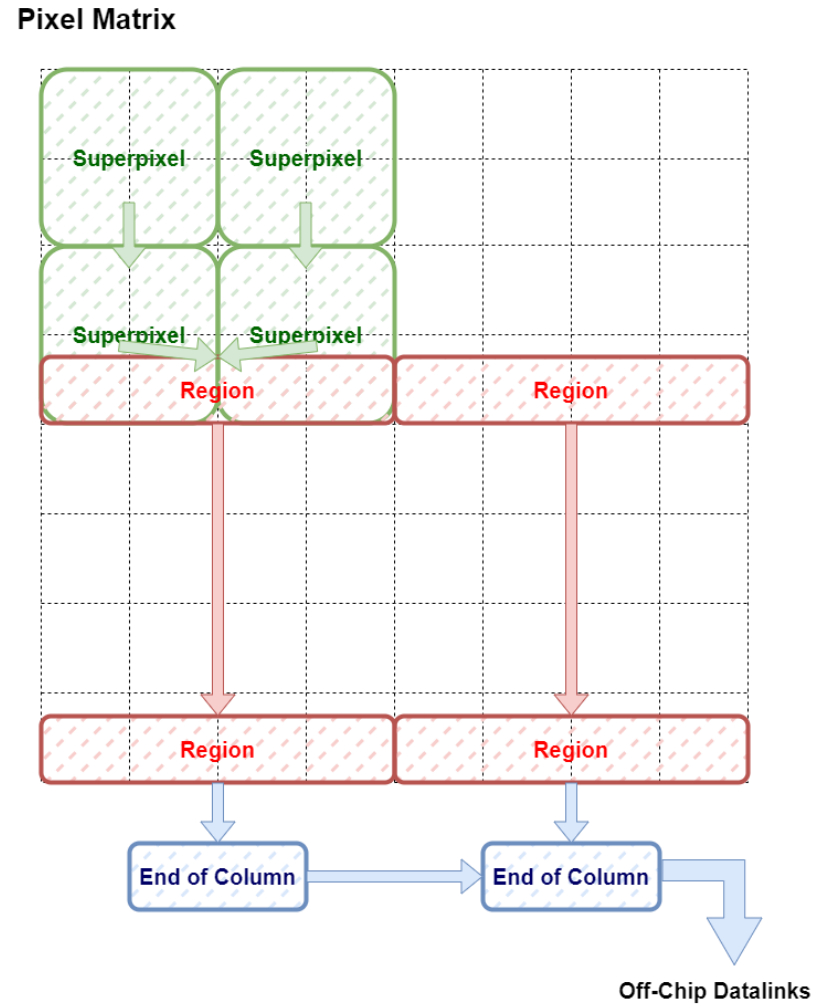
Model Velopix (VELO upgrade I ROC)

Simulate higher occupancy events

Find bottlenecks

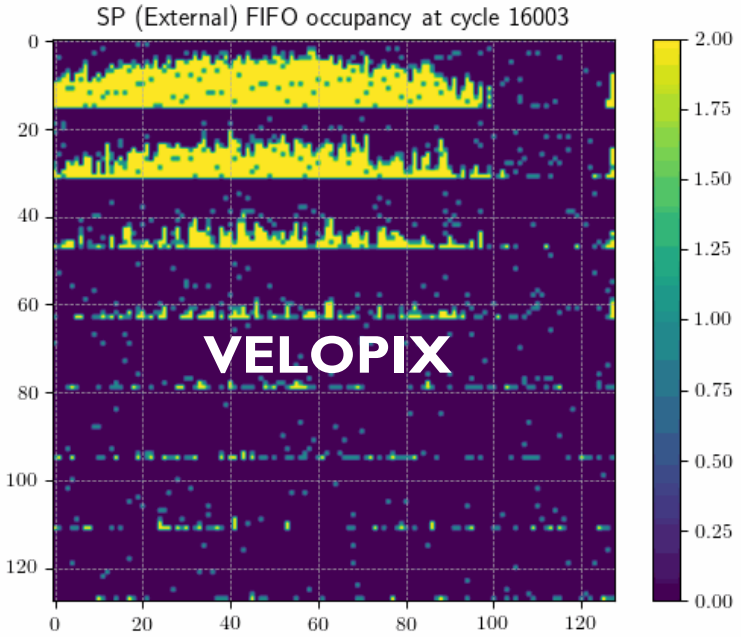
Optimize architecture

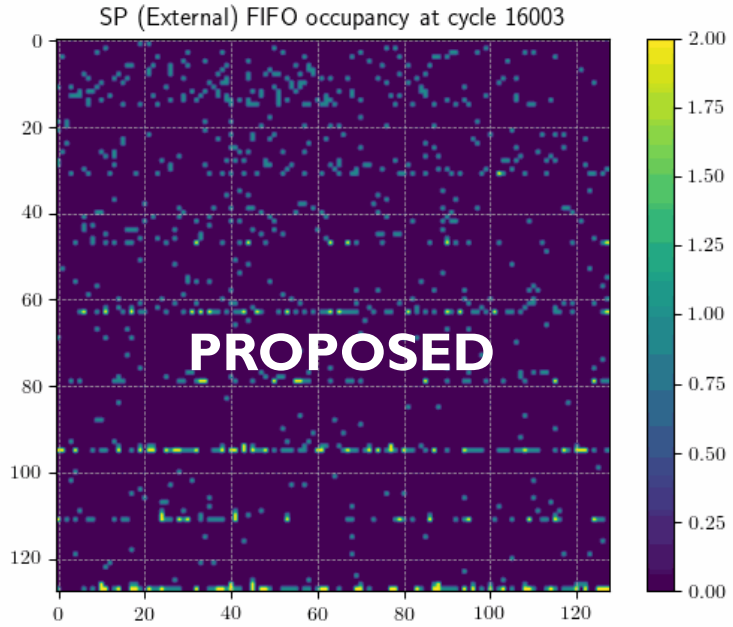
Repeat!



LHCb Velo upgrade II simulations

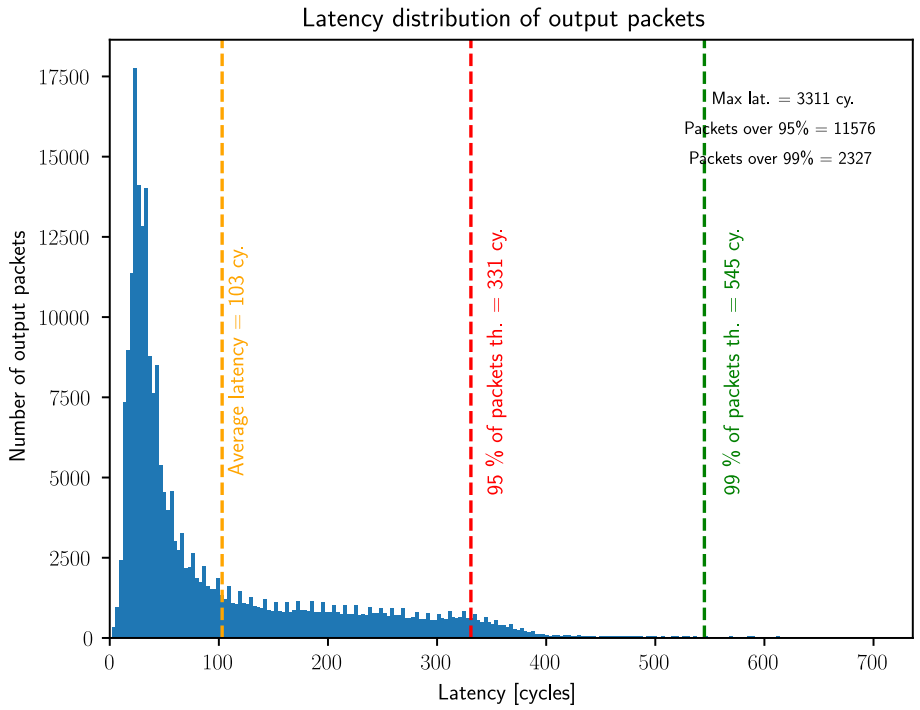
	Velopix	Proposed
Pixel	256x256	256x256
SP	128x128	128x128
Regions	64x8	128x8
EoC	64 (8 ch.)	128 (16 ch.)

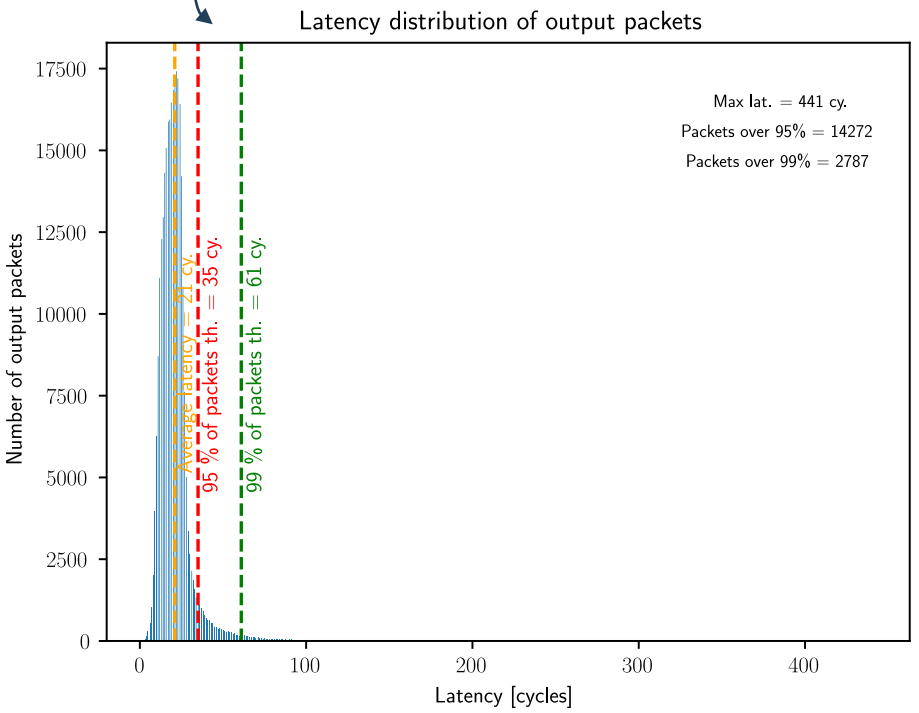




LHCb Velo upgrade II results

	Velopix	Proposed
Readout eff.	80 %	99.5 %
Avg. latency	~100 cy.	~20 cy.





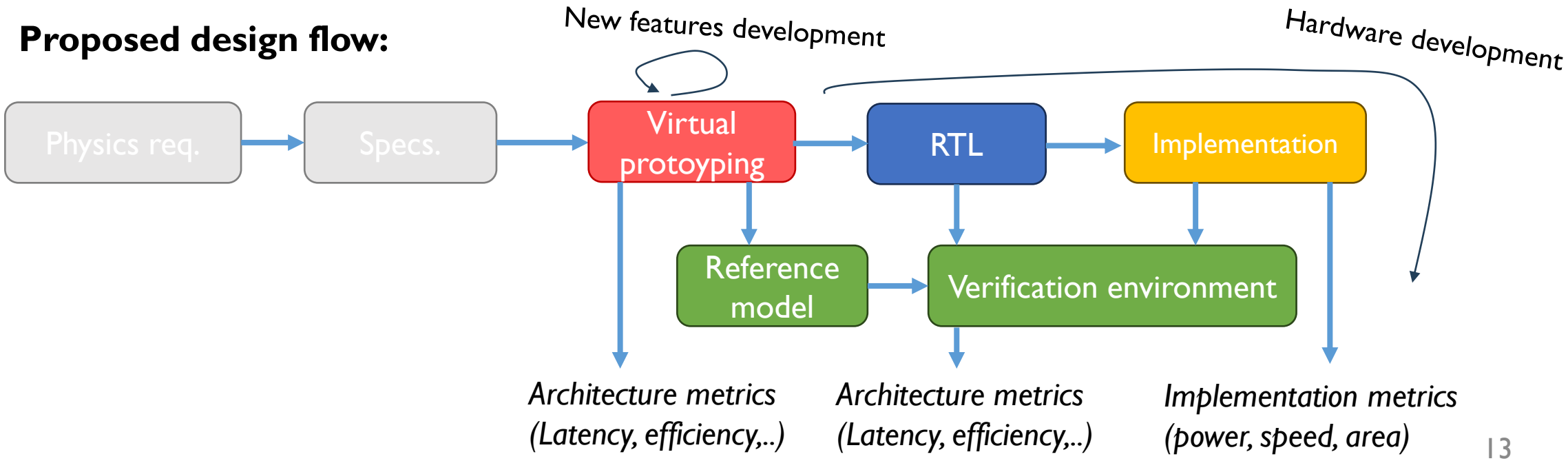
ON-CHIP PACKET SORTING FEATURE DEVELOPMENT

Use a **high-level** description of the system to:

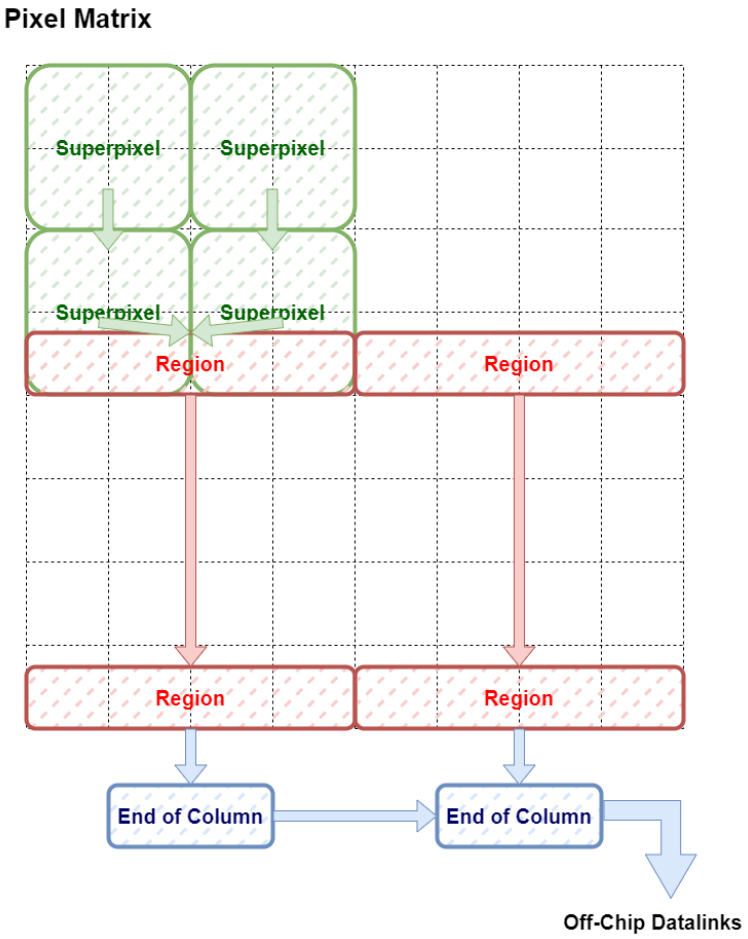
- Provide a self-contained prototyping environment
- Size the parameter of the new feature

Finally, design RTL and run implementation to get the full metrics.

Proposed design flow:



ON-CHIP PACKET SORTING



Problem: Data-driven readout provides out-of-order packets

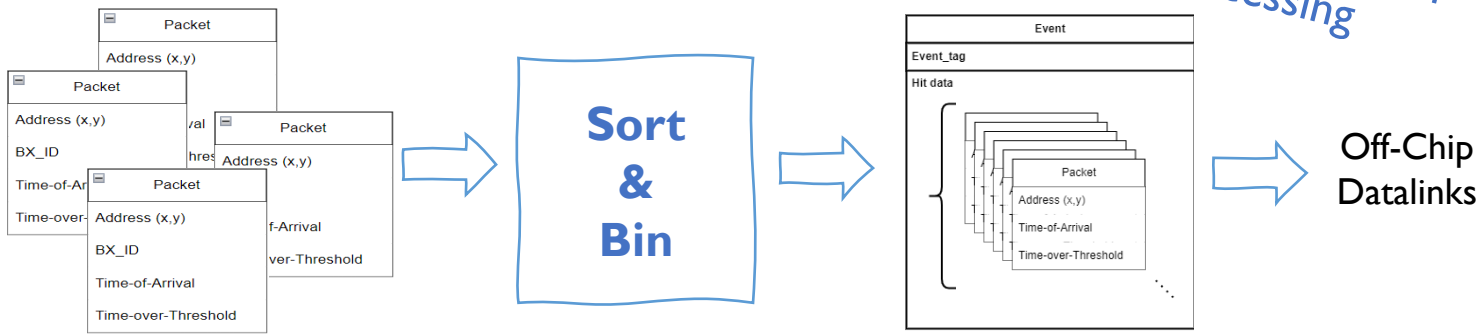
Proposed solution:

Sort&Bin module accumulates hit packets over time and groups them in bins based on event tag

Goals:

- Ordered packet readout
- Fixed latency
- Data reduction (~20 %)

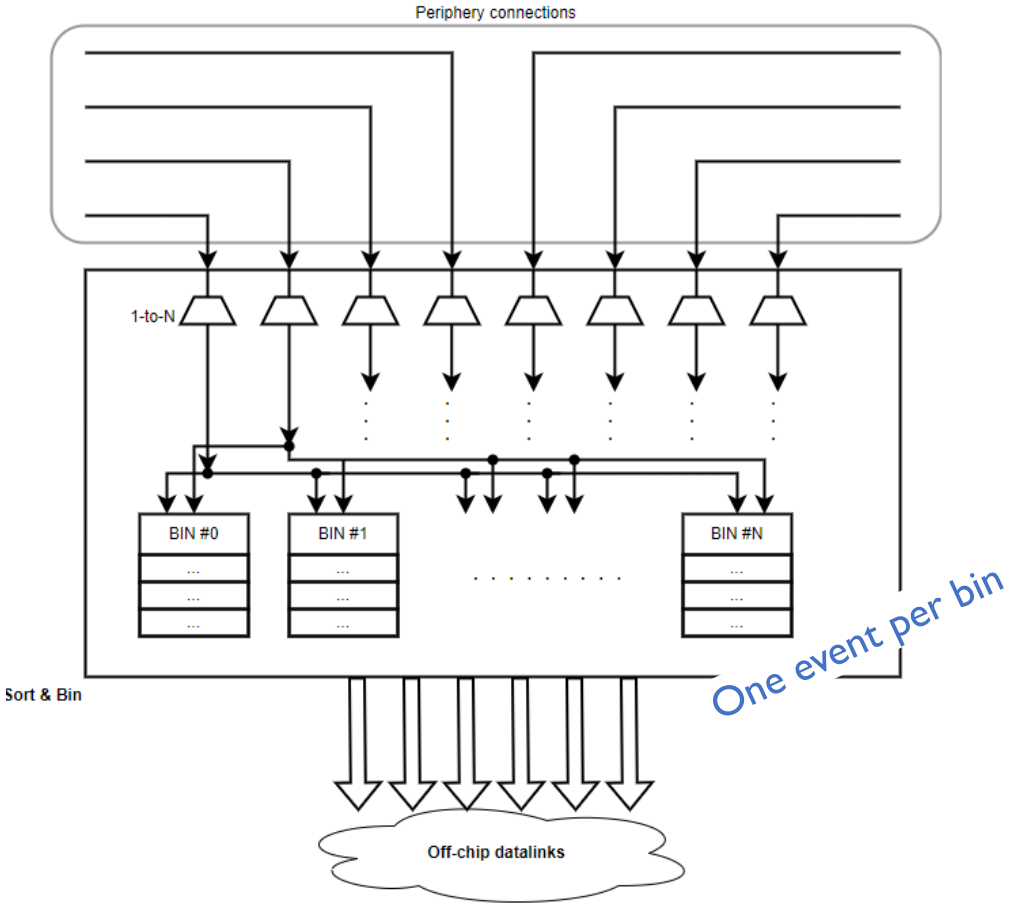
Enablers for on-chip data processing



Out-of-order hit packets

Event frame of hit packets

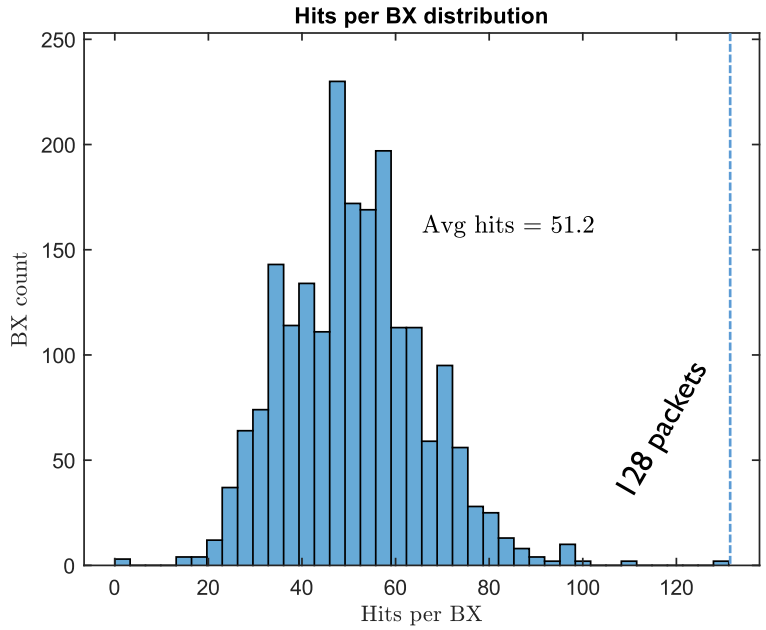
ON-CHIP PACKET SORTING DESIGN SPACE EXPLORATION



SystemC description of the module

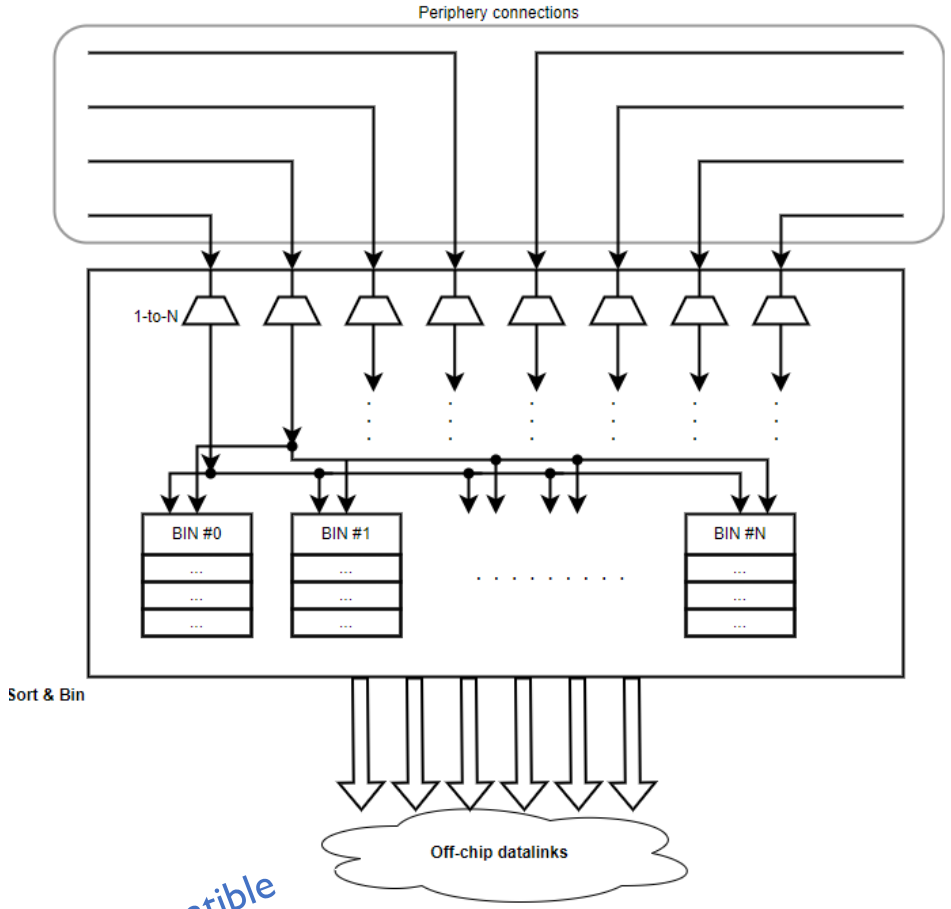
The bin size depends on the maximum number of hits (or clusters) per event

Bin size



Target: Bin size > Max packets per event

ON-CHIP PACKET SORTING DESIGN SPACE EXPLORATION

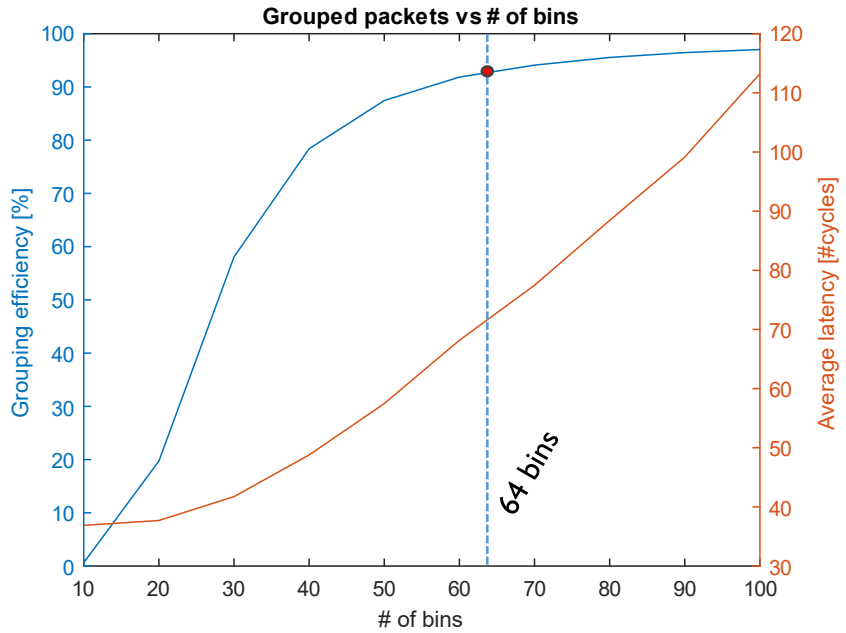


HLS compatible

SystemC description of the module

The number of bin depends on the latency of the readout efficiency and on the target grouping efficiency

Number of bins

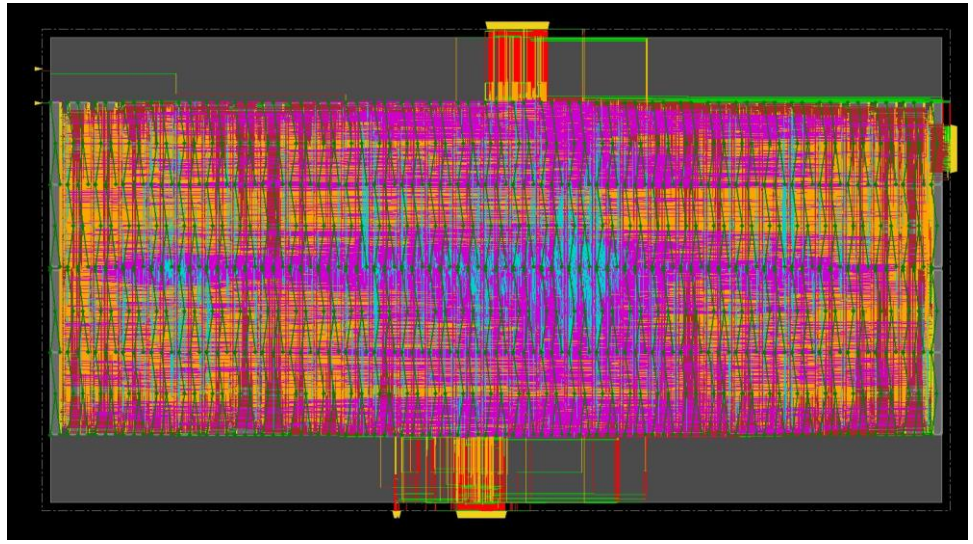


Target: >90% grouping

ON-CHIP PACKET SORTING IMPLEMENTATION

Sort&Bin RTL developed and verified against the model:

- **64 Bins** with 128 packets per bin
- Design based on Dual-Port foundry SRAM



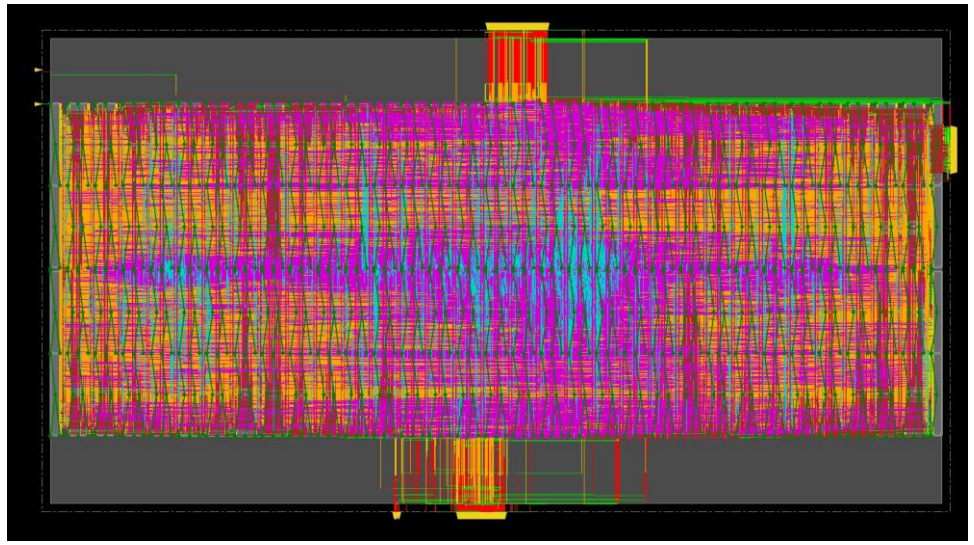
Implemented in 28nm bulk technology

- Close timing at 320 MHz
- Power consumption: ~50 mW
- Area: 3.6 mm^2

On-chip packet sorting implementation

Sort&Bin RTL developed and verified against the model:

- **64 Bins** with 128 packets per bin
- Design based on Dual-Port foundry SRAM



Implemented in 28nm bulk technology

- Close timing at 320 MHz
- Power consumption: ~50 mW
- Area: 3.6 mm^2

Should fit in a pixel ROC periphery and consume less than a 25 Gbps serializer

Summary

PixESL is a valid tool for high-level modelling and virtual prototyping.

The framework presents an effective and quick approach:

- Workforce: ~1 fellow and 1 student
- Development time: ~3 months
- Modelling of Velopix: ~1 month
- Modelling of Sort&Bin: ~1 week
- Runtime: ~15 BX/s

Future outlook

Development started in March 2023:

- Tool is in an early maturity stage
- More use cases and real applications help us target and review the development.

Model development is still ongoing:

- Cycle vs Event-based
- Test-bench generation
- Power estimation tool
- Front-end modeling

A few already in the pipeline

The release is expected for Q3-Q4 2024,

but we are open to developer or user direct collaboration.

EP-ESE
ELECTRONIC SYSTEMS FOR EXPERIMENTS

THANKS!

DAVIDE.CERESA@CERN.CH