Platform-Based Design: The Next Reuse Frontier

Bob Altizer

VSIA PBD Study Group

BASYS Consulting

Why Platform-Based Design (PBD)?

Platforms are the reuse paradigm for the next level of issues VSIA is facing, with important interoperability issues and time-to-market reductions expected.

- Platforms are the catalyst for systematic reuse, rapid development, and integration of derivative products, and can drive dramatic reductions in time-to-market.
- The effective use of platforms throughout the supply and design chain requires fundamental agreement on foundations and definitions, and enables better interaction with suppliers and customers.
- It's an integration thing -- integration skills and design flow are vital.
- VSIA members have more to gain by collaboration than by keeping all their work in PBD proprietary.
  - We're desperately in need of a common vocabulary and agreed definitions
- PBD represents both a set of types of *implementation styles* (e.g., fixed HW, reconfigurable HW, SW-based, etc.), and a set of *application domain-specific types* (e.g. network processors, automotive, wireless, set-top boxes, etc.).

PBD SG/DWG Draft Charter

- VSIA believes the embedded systems industry can reduce risk and cost, and enhance time-to-market and return, by applying principles of platform-based design for families of products.
- At present there is no standard, repeatable approach to defining, scoping, specifying, or reusing a platform. The PBD Study Group and follow-on DWG will lead industry standardization of platform engineering methodology for SoC-based embedded systems.
- While platform-based design is applicable at many levels of integration within a product, our scope will be limited to the SoC level and below.
- Today's platform is tomorrow's component!

PBD Study Group Membership

• Carl Pertry	Alcatel
• Bert Aerts	Alcatel
• Frank Pospiech	Alcatel
• Srini Rao	Analog Devices
• Frederick Tarverdians	ARC Cores
• Ian Phillips	ARM
• John Goodenough (*)	ARM
• Carlos Oliver-Yebenes	ARM
• Bob Altizer (*)	BASYS
• Grant Martin	Cadence
• Thomas Harms	Infineon
• Takahide Inoue	STARC
• Homer Hegedus	Motorola
• Jauher Zaidi	Palmchip
• Heinz-Josef Schlebusch	Synopsys SLD
• C. Y. Pei	Toshiba
• Kun-Bin Lee	NCTU
• Jim Tobias	JRTobias
• Abhijit Sarid	GDATech
• Mike Kaskowitz	Mentor Graphics
• Angela Sutton	Mentor Graphics
• Anssi Haverinen	Nokia
• Joachim Kunkel	Synopsys
• Mason Weems	Thymetech

* co-chairs

So What's a Platform?

Figure 1. i250 Platform Chipset Block Diagram

Definition: What is a Platform?

- There seem to be as many definitions as the number of people asked!
- Some theoretical definitions beginning to emerge:
  - "A ***platform*** is, in general, an abstraction that covers a number of possible refinements into a lower level. For every platform, there is a view that is used to map the upper layers of abstraction into the platform and a view that is used to define the class of lower level abstractions implied by the platform" (Alberto Sangiovanni-Vincentelli, GSRC white paper on Platform-Based Design)
  - "An ***integration platform*** is a reuse mix-and-match environment designed specifically to target an application domain. The domain is selected based on market objectives and is focused to yield a high probability of reuse over a target period of time." (Cadence white paper "The IP Reuse Evolution")
  - "A ***platform*** is a collection of assets which can be used to leverage reuse and rapidly develop new products. At a minimum, it defines the operating environment, high level product architecture for all products developed based on this platform, and set of development policies for extending the platform and developing point products from the platform." (Motorola PCS/ATSO "Reuse Lifecycle Model, v1.0")
  - "An ***embedded system platform*** is an architectural framework for rapid integration of embedded SoC-based designs, consisting of a set of pre-qualified software and hardware IP blocks and a methodology to support rapid architectural exploration, integration, and verification." (Frank Pospiech, Alcatel)

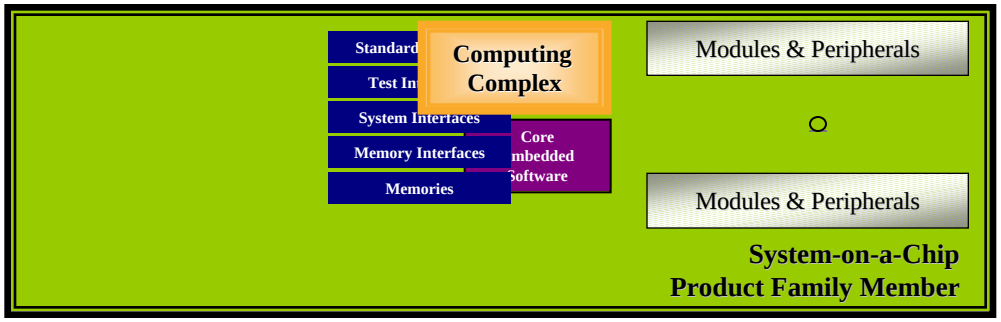
Definition: What is Platform-Based Design?

- If a ***platform*** is a reliable technology on which an implementation can be constructed (IP) …

  … then ***platform-based design*** (PBD) is a technology which presents a reliable and quantifiable translation from a modeled behavior to an implementation behavior.
- At a concrete level, many SOC platforms and their characteristics, and issues in platform-based design, have been identified by Embedded Systems Study Group members
- Motorola, Philips, and Infineon concepts follow...

Platform Postulates

- There exist sets* of functional components (platforms) that can serve as the basis for multiple derivative SOC products (*aggregations of IP)



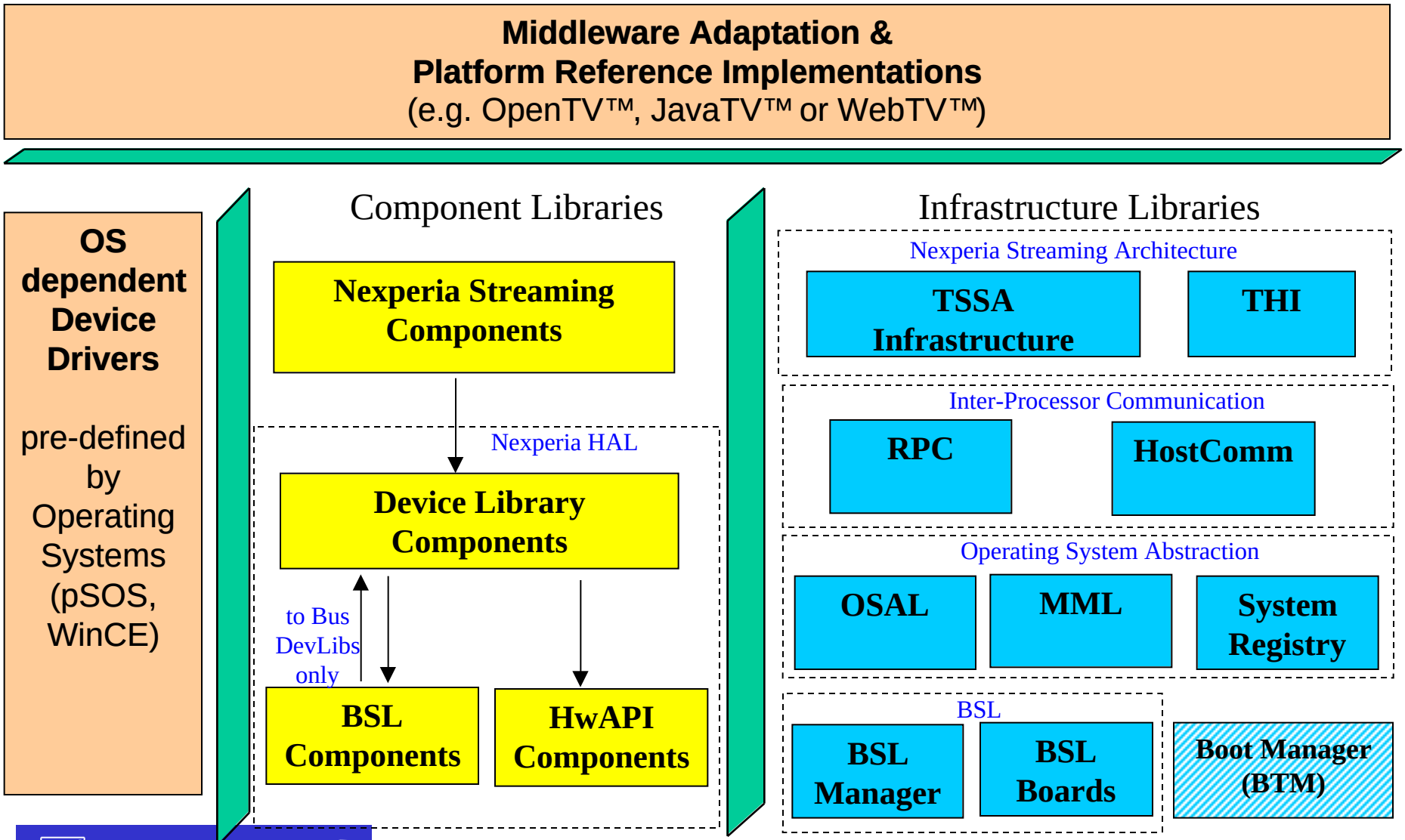
- Platform-based development depends on the existence of:
  - defined and verified platforms
  - an integration-oriented design flow
  - integratable market-specific IP components
  - tool, application and systems support
- Economic advantage from derivative products will be realized through:
  - enhanced product capability
  - improved margins
  - improved time to market
  - improved quality
- Advantages will be large enough to justify investment in development of platforms, collateral IP, tools, and support

intelligence everywhere

MOTOROLA digital dna

VSI Alliance

Nexperia Software Module Architecture



High Level Architectural View

- The high level architectural view shows what the SOC looks like to a piece of code. In a multicore SOC we need the view from each core.
- These software architectural issues become visible at the Platform level
  - Address maps; bridges
    - Base address for each peripheral
    - automatic configuring; querying of the peripheral space for peripherals
  - Interrupt domains
  - Boot procedure (per CPU and/or for whole SOC)
  - BSP/PSP for chosen RTOS
  - Chip-level configuration:
    - PLL programming; clock domains
    - SCU: System Control Register
    - BCU: Bus Configuration Register
    - Registers to program access details to external memory

Nick Ray
Infineon

The Study Group's Working Definitions

- **Platform –** A library of virtual components and an architectural framework consisting of a set of integrated and pre-qualified software and hardware IP blocks, models, EDA and software tools, libraries and methodology to support rapid product development through architectural exploration, integration and verification.
- **Platform Based Design –** An integration-oriented design approach emphasizing systematic reuse, for developing complex products based upon platforms, intended to reduce development risks, costs, and time to market.

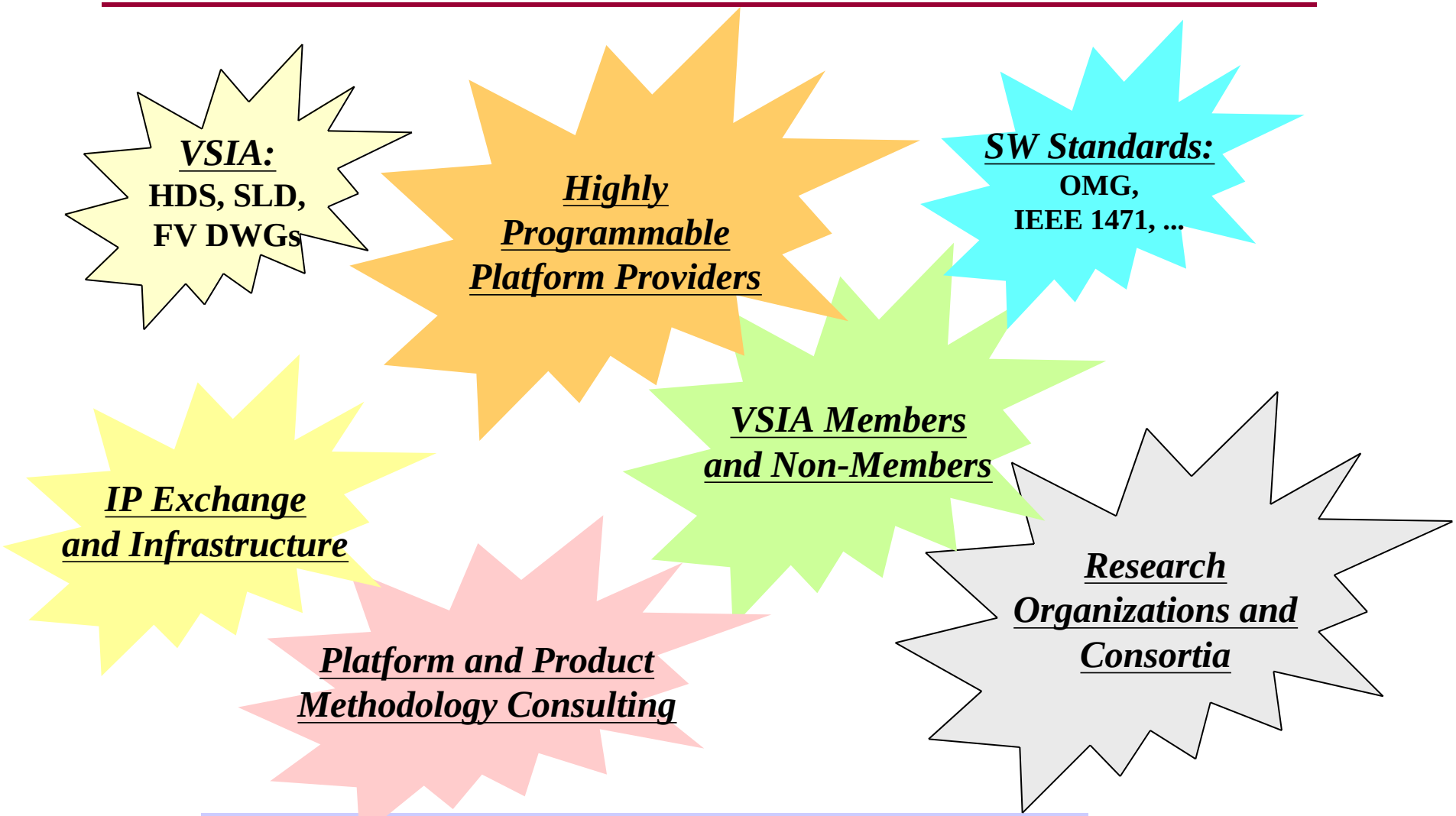
Within the VSIA the scope of the platform is the SoC.

Today's platform is a virtual component for tomorrow's platform!

Platform-Based Design Needs...

- The industry needs consensus on PBD ...
  - Definitions & taxonomy
  - Structure and options
- For each taxonomy, platforms will have a core set of things in common, and support sets of differences
  - Commonalities must be clarified for understanding
  - Commonality in platform; variabilities across product family members
- A search for ...
  - Commonality in methodologies and interface definitions
  - Layering
  - Product family economic models
  - Modeling requirements
  - Integration-oriented design flows and methodologies
  - Technology adoption & transfer, and design productivity impacts
- From the VSIA Embedded Systems Study Group, there is no shortage of expertise or interest!

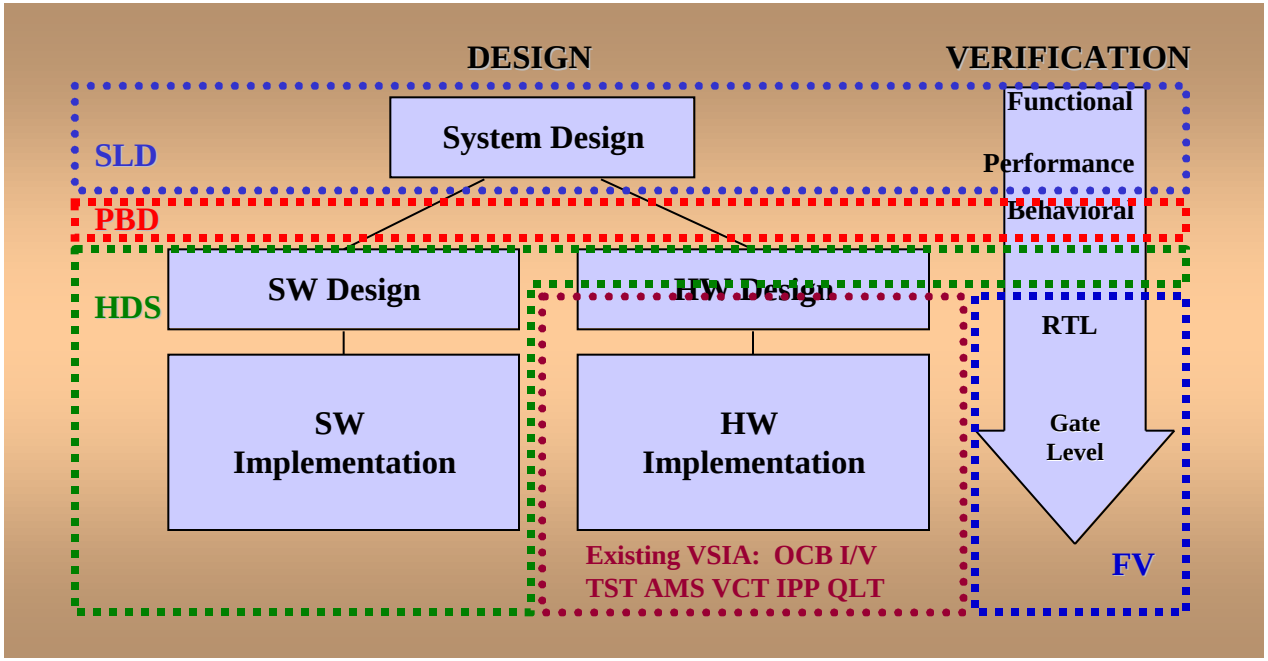
Possible Additional PBD Partners and Collaborators



Many individual areas are covered, but no one is tying them together.
VSIA can do this job!

Relationship Between VSIA DWGs

- When VSIA was first formed, almost everything covered by the roadmap was concerned with RTL and below.
- A system level design (SLD) group was established with the notion that in a few years there might be something above the RTL level.



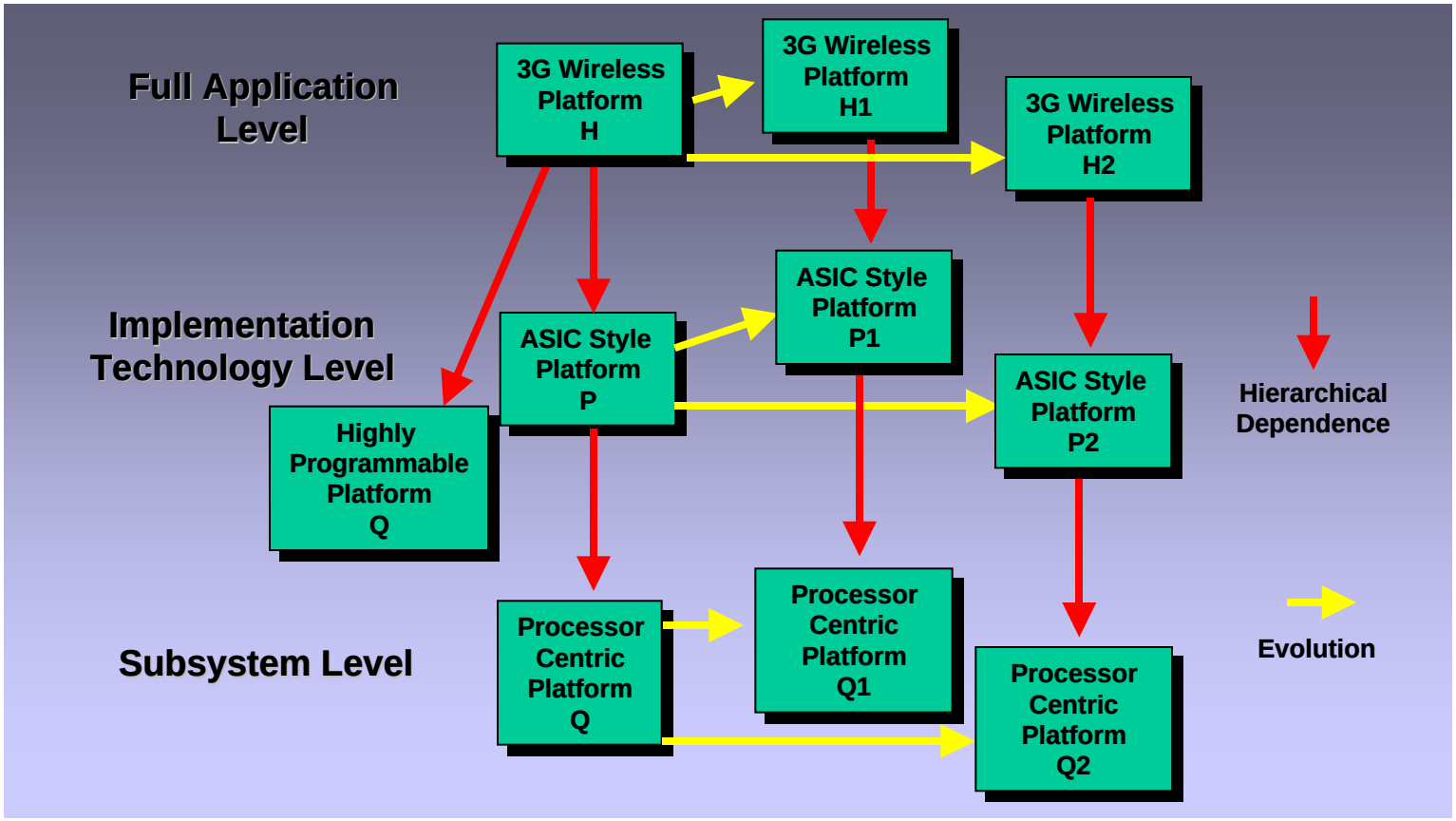
- Today, VSIA addresses a number of groups above the RTL level
- The Technical Committee (TC) is the top-level architectural authority, assuring the role and charter of each group is clear to them and to the external world, and each group's work is leveraged with the others to the greatest degree.

Agreement with the VSIA TC

- Platforms address the bigger picture:
  - Embedded software
  - Integration-based development flows
  - Connectivity
  - Verification
  - System-level modeling
  - System-level design intersects a portion of platforms
- Charter, taxonomy, and vision are the key deliverables
  - Coordinated charter and "Grand Unified Taxonomy" with HDS, SLD, and FV DWGs
  - Characteristics in common across all taxonomies
  - Include characteristics and categories of existing platform approaches
- PBD activities will be undertaken with "project orientation"
  - Define finite tasks to which members can be assigned – and be accountable
  - Plan for task completion in 6 months or less

PBD Points-of-View: Hierarchy & Evolution

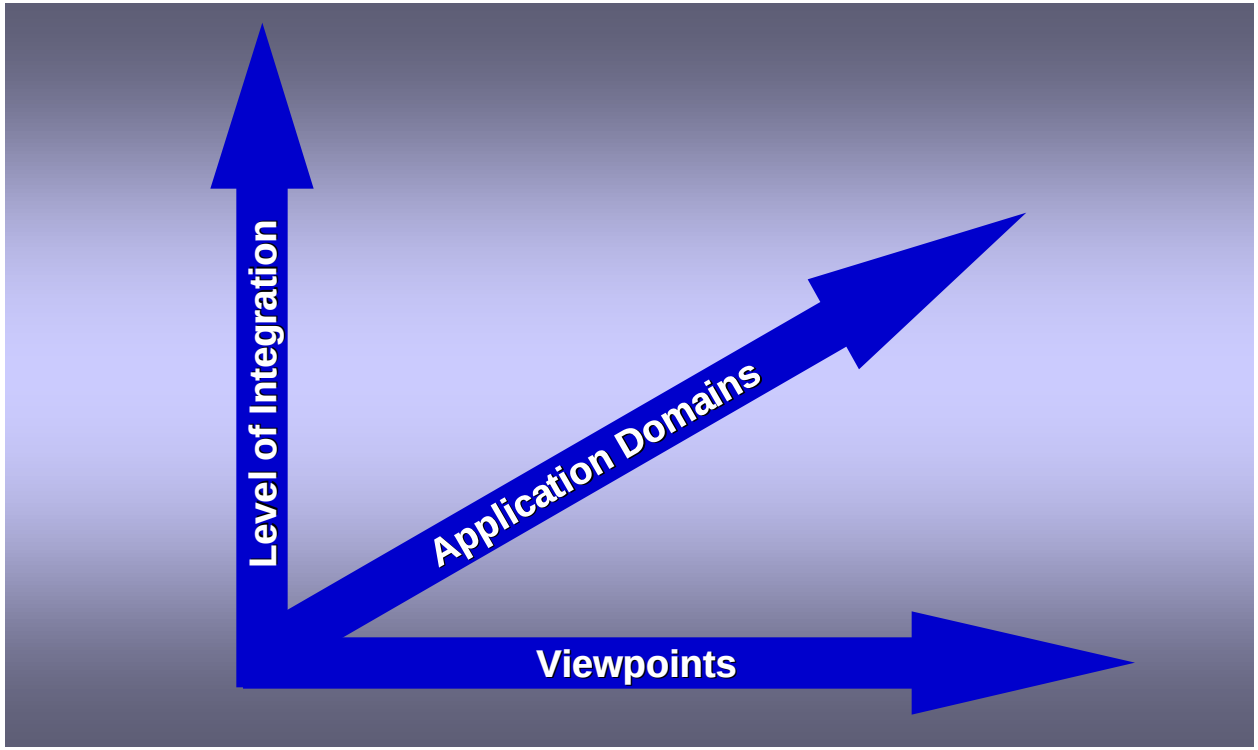
- Multiple levels of platforms exist within any product or system, with at least some degree of hierarchy between levels (though that hierarchy is probably imperfect) and evolution across platform generations.



Source: Carlos Oliver-Yebenes, ARM

PBD Points-of-View: Platform Types

- There is not just one kind of platform; we have to consider several independent issues:
  - Level of Integration (within the overall product)
  - Viewpoints (the concerns of some stakeholder)
  - Application Domains (wireless, industrial, networking, automotive, etc.)



PBD Points-of-View: Platform Types

- Some as-built examples exist describing patterns or types of platforms (e.g., full application, processor-centric, communication-centric, and reconfigurable platforms); others can probably be identified as well. Some of these may be linked hierarchically.

Platforms	SW components	HW components	SLD support	SW devt Support	HW devt support	Logistics
Personal comm-unicator	Application OS	Cellu Engine, mechanics, accessories	UI emulator	Application cellu API, accessory API, game API, etc.	Front plate/ battery rules	Manufacturing
Cellular engine	L2/3 SW	Cellu BB, RF, DRAM, mic, LCD, speaker	L2/3 stub	OS hooks	Mechanical constraints	Component sourcing
Cellular BB	modem L1 SW	Processor, custom modem logic	L1 SDL stub	L1 API, peripheral/ processor HAL	PCB data	P&R, proto relationship
Processing subsystem	RTOS	µC core, DSP core, I/O, cache, flash	Processor C models, bus analysis tools	Compiler, debugger, HW API	Si library, back-end tools, bus generators	Design rules, process/design data availability, supported EDA

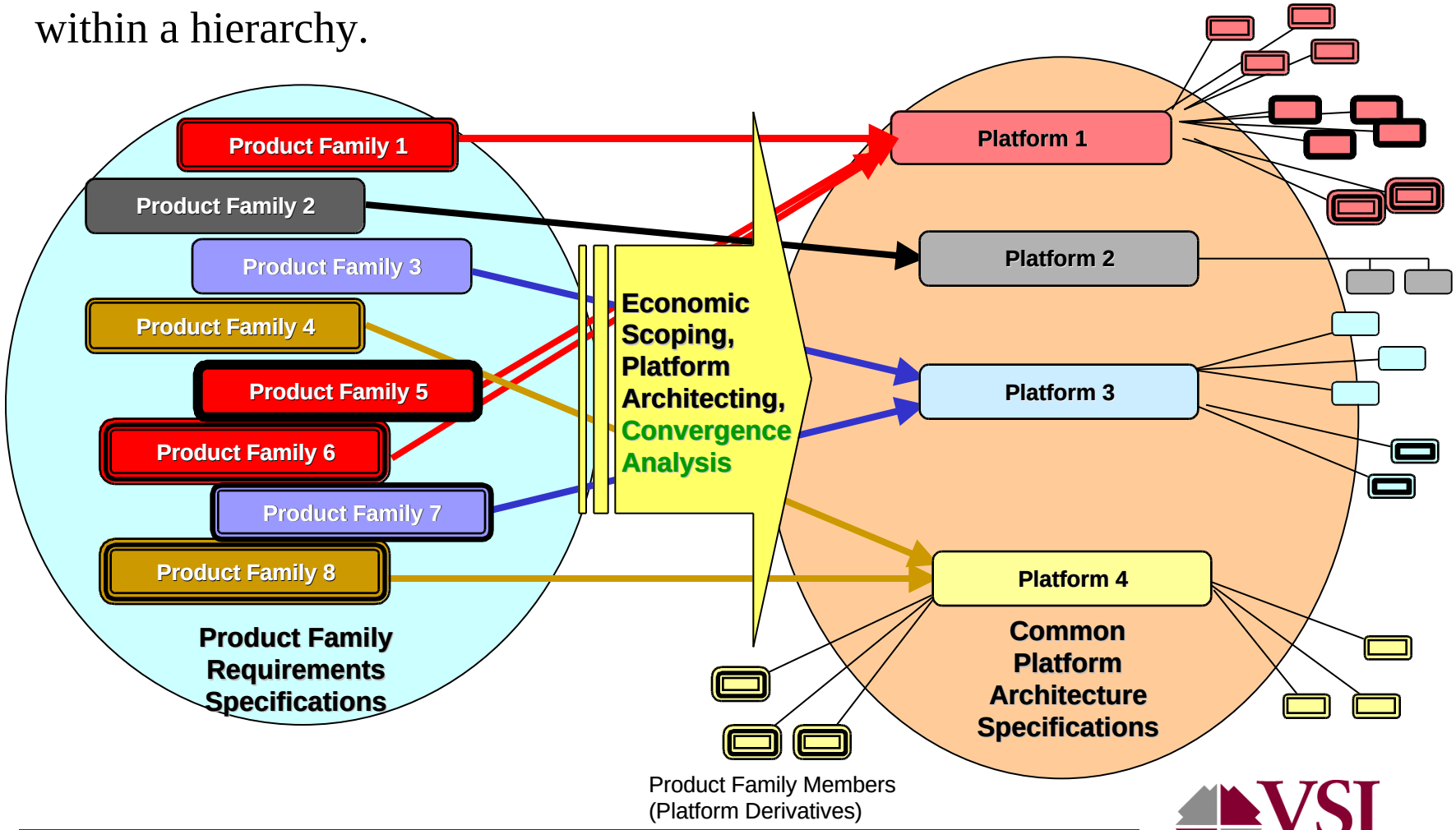
Source: Anssi Haverinien, Nokia

PBD Points-of-View: Implementation, Tools & Models

- A platform at any level exists as a black box, with:
  - a real **implementation** (an integrated, verified set of both hardware and software functional components);
  - a definable, complete **architectural description** (may be derived from an as-built implementation);
  - a complete and accurate set of **models** describing its actual behavior (this may be redundant with the AD, or the AD may call for more models than yet exist);
  - a set of **tools** to permit integration of the platform model into the model of a higher level system;
  - a set of **tools** to permit integration of the real platform implementation into the implementation of a higher level system.

PBD Points-of-View: Top-Down Analysis

Product family-based, top-down analysis is a promising approach for defining requirements and scoping of a platform at any particular level within a hierarchy.



PBD-SG "Project" Recommendations

To be done within the scope of the PBD Study Group:

- **Charter:**
  Draft a charter for the succeeding PBD DWG.

- **Glossary/Taxonomy:**
  Define terms related to platforms and platform-based design for inclusion in VSIA top-level Grand Unified Taxonomy (including PBD, HdS, SLD, and FV).

- **Multiple levels**:
  Specify a few lower levels of integration platforms we want to deal with, up to the level of an SOC, and describe the nature of hierarchy, if any, from lower levels to higher. Identify any attributes that can characterize a level or that may be incorporated by platforms at higher levels.

- **Patterns**:
  Identify a reasonably full set of existing platform types (up to 8 to 10), and show how they relate: hierarchically, as peers, or whatever.

PBD-DWG "Project" Recommendations

To be done by the follow-on PBD DWG:

- **Implementation, models, and tools**:
  Identify the elements of platform architecture needed to represent a real implementation of each level type of platform (this may consist of the set of architectural views appropriate for each platform, which implies the set of models needed to represent the platform as a whole). Describe the processes for integrating a platform model into a higher level model and a real platform implementation into a higher level implementation; this implies the tools that will be used.

- **Architectural Views:**
  Define standard views in platform architectures; may be possible to aggregate into a standard sets for a particular platform types.

- **Top-down analysis**:
  Consistent with the identified levels, platform types, architectural views, models, and tools, describe a top-down analytical process that can specify the proper scope and recommended architecture of a platform, based on the nature (features, attributes, requirements, etc.) of higher-level systems (i.e., products) into which it will be integrated.

PBD Study Group Proposed Major Milestones

12/2001: Kick-off

1/2002: Start working meetings and email exchanges

- Develop initial consensus on PBD issues and definitions
- Develop joint charter with HDS, SLD, and FV DWGs
- Start draft of PBD Taxonomy

3/2002: eSoC presentation and draft Taxonomy unveiling (today) (in conjunction with HDS, SLD, and FV DWGs)

4-5/2002: Continue working meetings

- Finalize first release of Taxonomy
- Define charter for follow-on PBD DWG
- Identify and write white papers on recommended issues for follow-on PBD DWG

6/2002: PBD Taxonomy v1.0, PBD DWG charter and recommendations ready for VSIA BOD review (PBD-SG disbands, PBD-DWG continues)

PBD Study Group Conclusions

- We've gotten off to a good start
  - Defined our key issues and schedule
  - Prioritized our initial and follow-on work
- We're just blazing some much needed trails
  - Lots of discussion remaining on taxonomy and hierarchy, patterns
  - Establishing overall architectural consistency between PBD and the other related VSIA work in HDS, SLD, and FV
  - Some very challenging projects defined for the PBD DWG!
- Plenty of room for collaboration by partners interested in joining the PBD DWG … will we see you there?

For More Information, Contact the PBD Co-Chairs...

Bob Altizer

BASYS Consulting

bob.altizer@basysconsulting.com

+1-480-705-6976

John Goodenough

ARM Sheffield Design Centre

john.goodenough@arm.com

+44-114-282-8018