Register Management of a Complex Multi-Processor Based SoC

Dave Murray, Brian Clinton, Zoltan Sugar - Duolog

duolog technologies

www.duolog.com

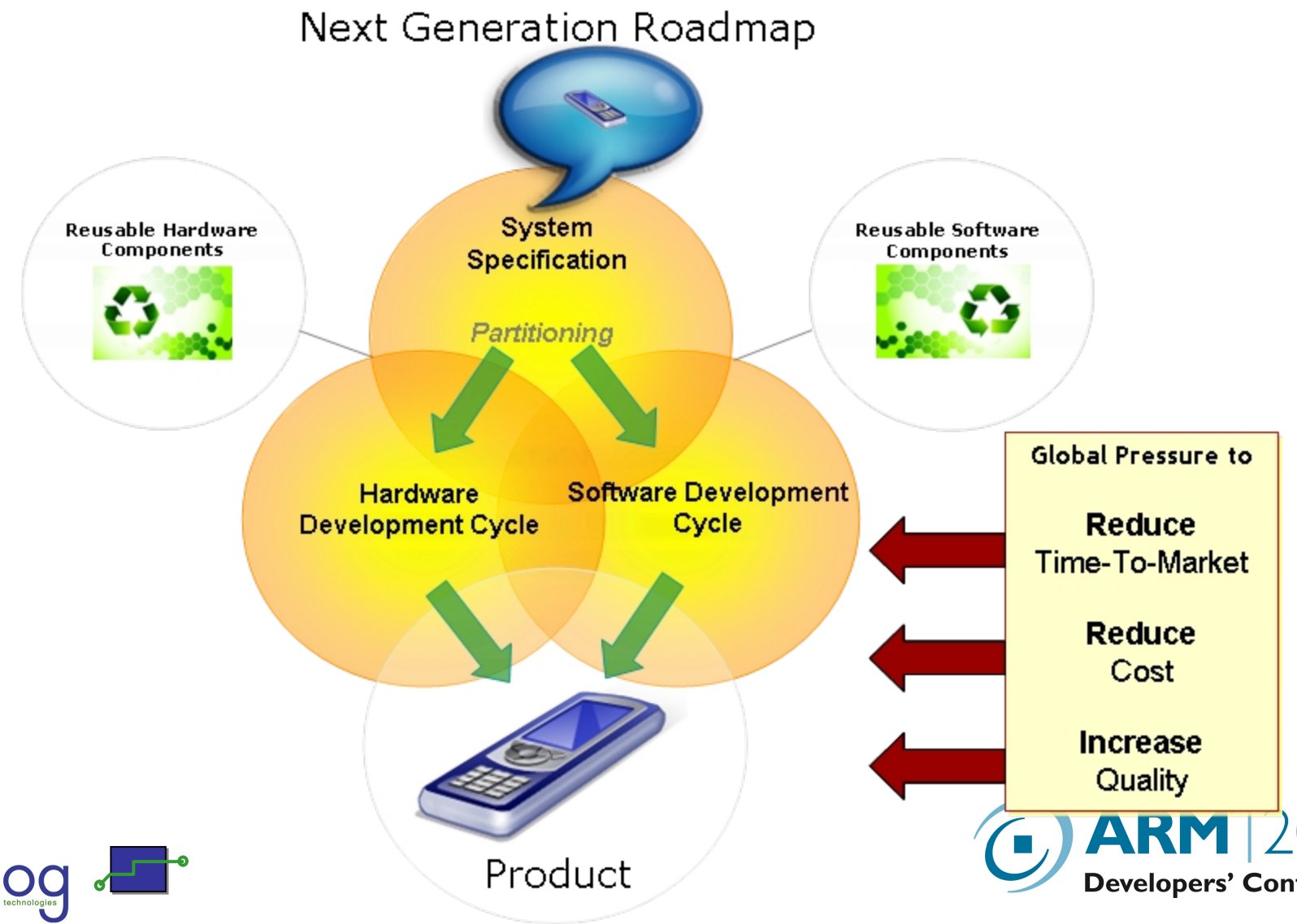
ARM | 2008 Developers' Conference

Example : OMAP3430

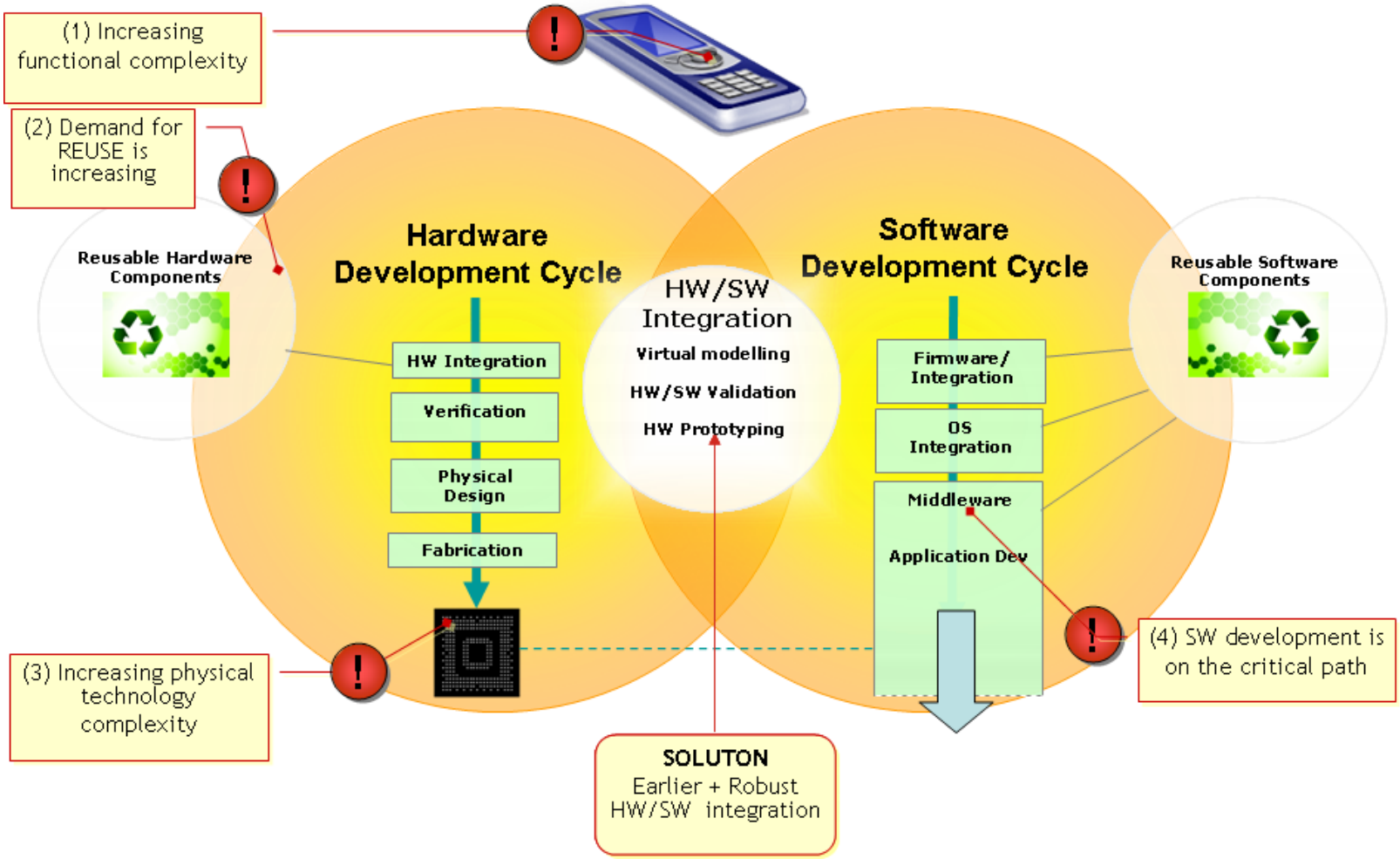
- Industry's first processor with advanced Superscalar ARM® Cortex™-A8 Core
- Industry's first processor designed in 65-nm CMOS process technology adds processing performance
- IVA™ 2+ (Image Video Audio) accelerator enables multi-standard (MPEG4, WMV9, RealVideo, H263, H264) encode/decode at D1 (720x480 pixels) 30 fps
- Integrated image signal processor (ISP) for faster, higher-quality image capture and lower system cost

Source : http://focus.ti.com/pdfs/wtbu/ti_omap3family.pdf

Multi-Processor Based SoC



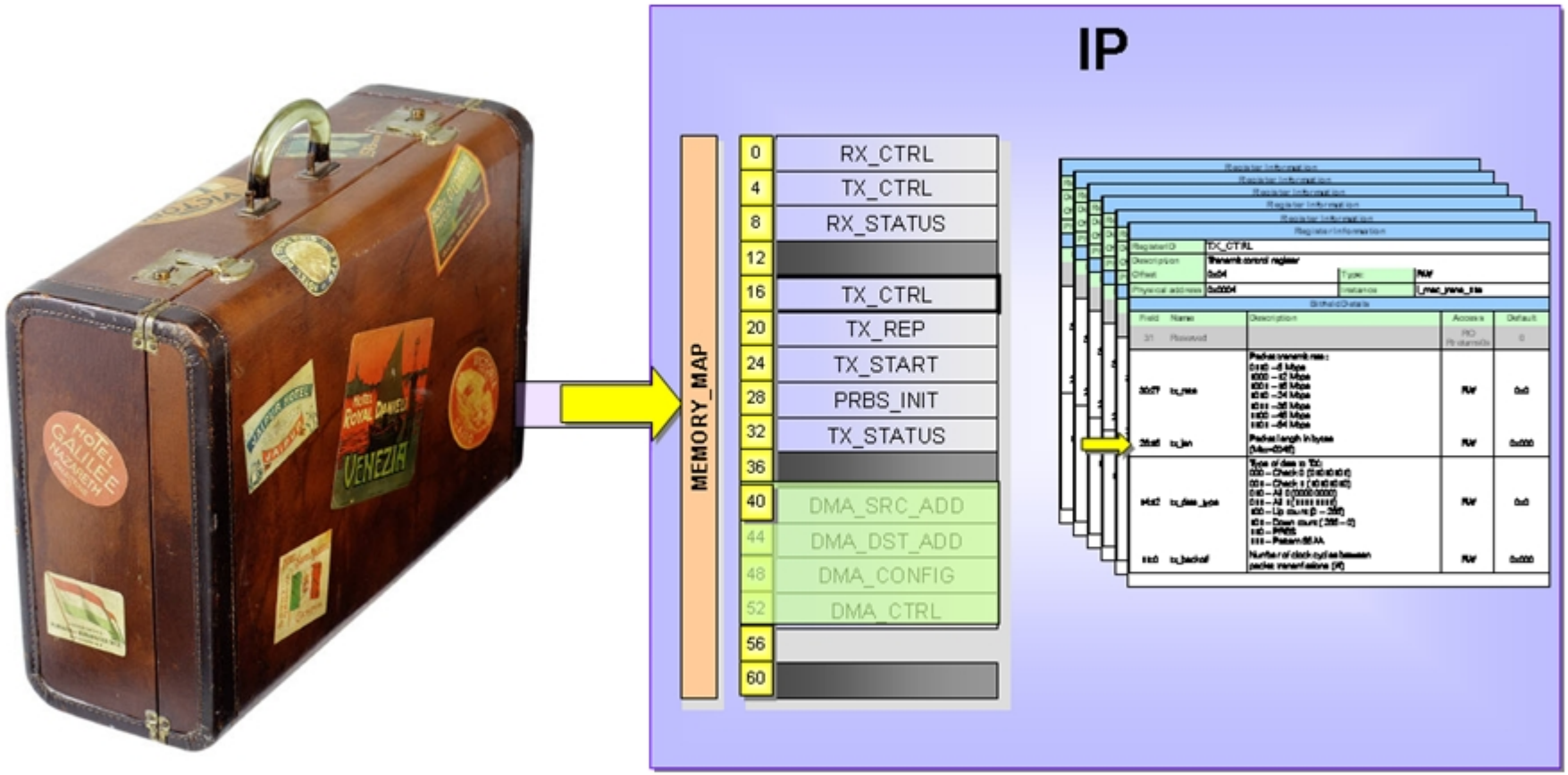
SoC Integration – Evolving Pain



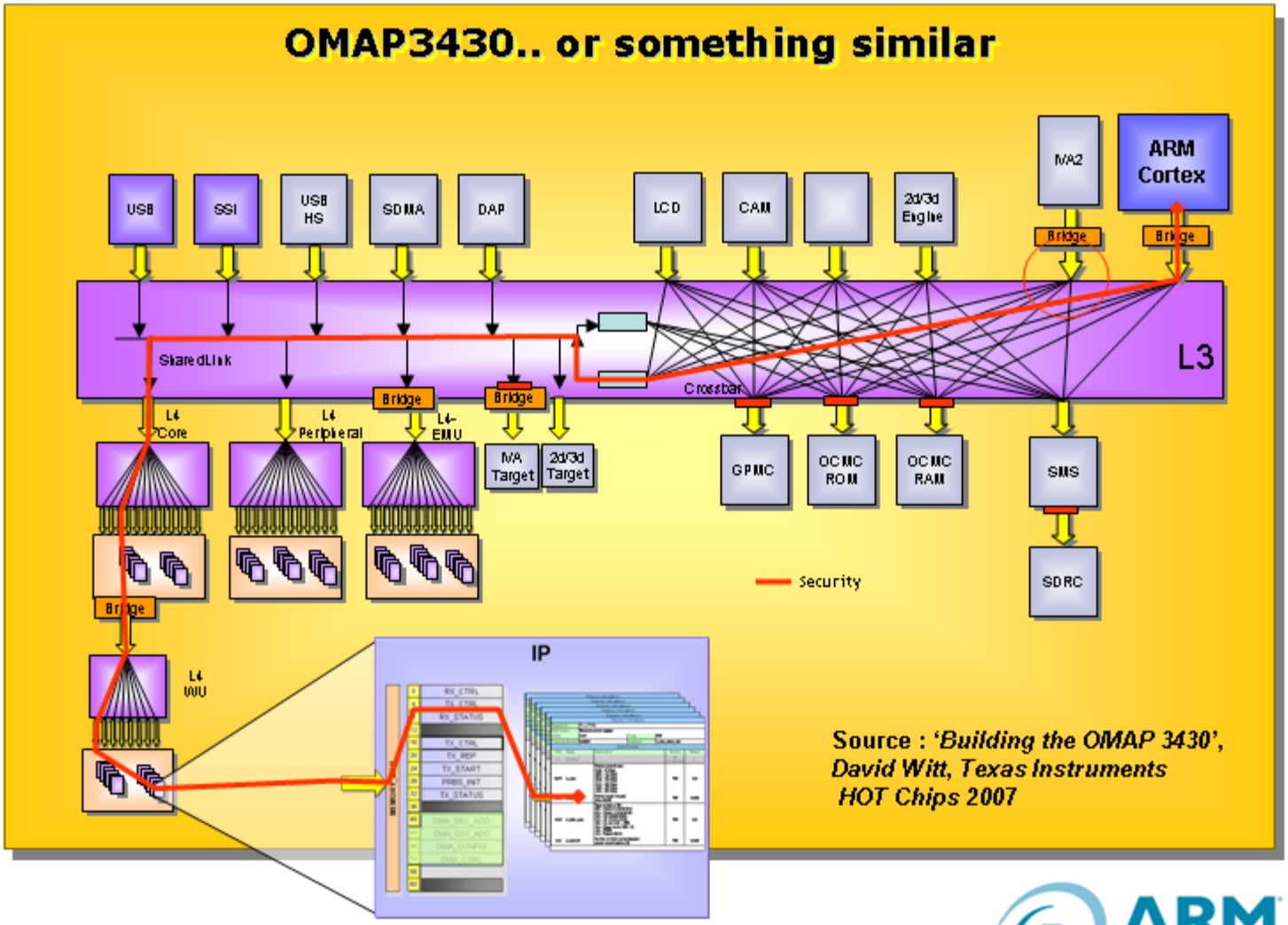
HW/SW inteface

Sometimes due to it's late arrival,
it will completely mess up your schedule

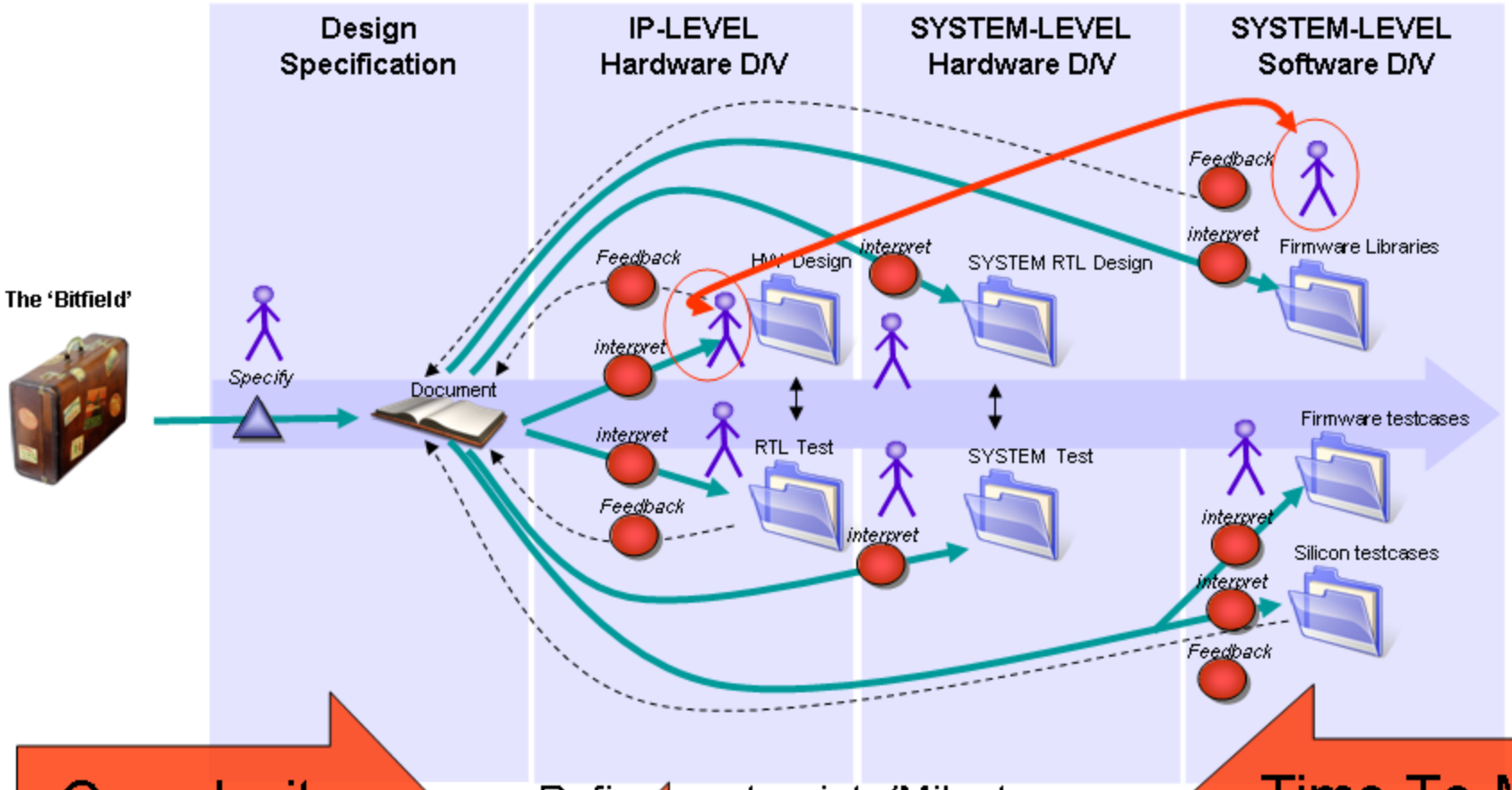
Register and Bitfields



SoC Complexity



Traditional 'Baggage Handling'



Complexity

Refinement points/Milestones

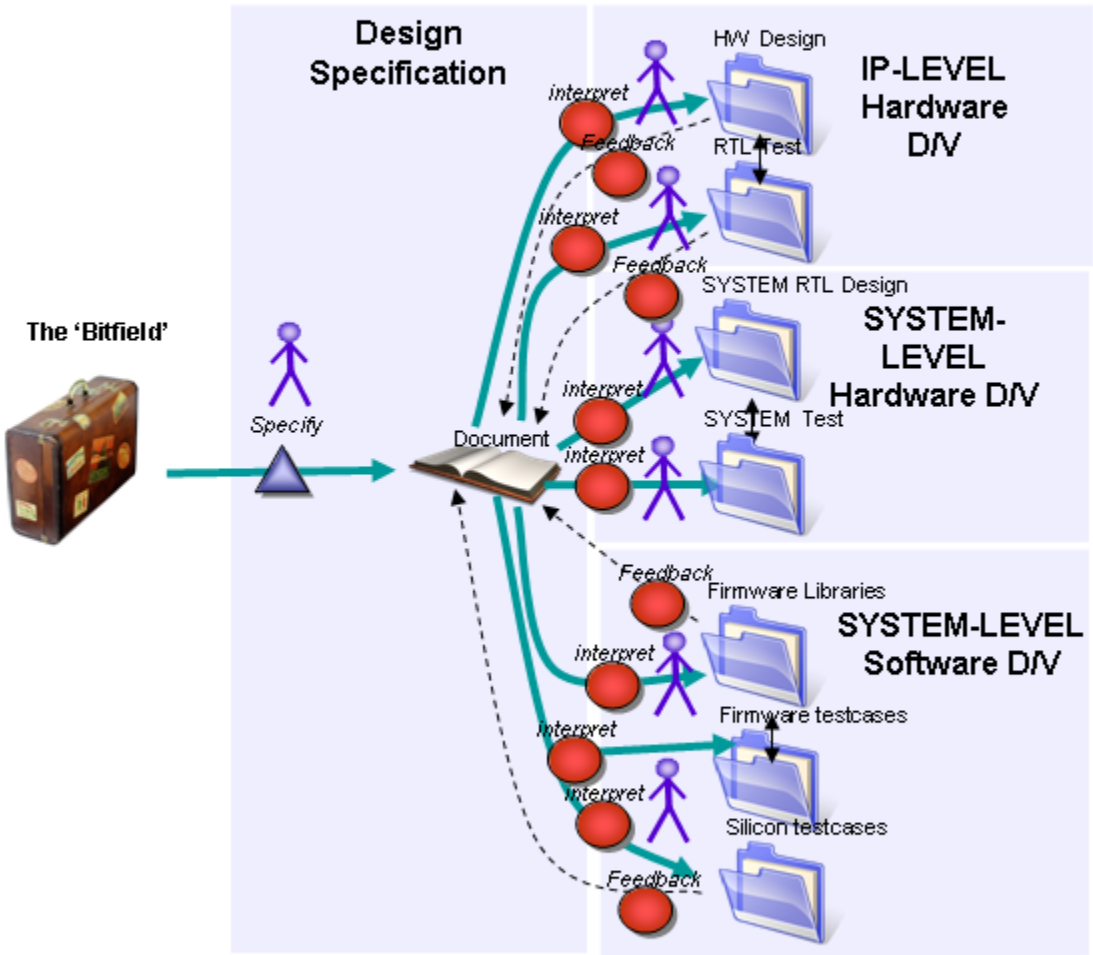
Time-To-Market

Quality

duolog technologies

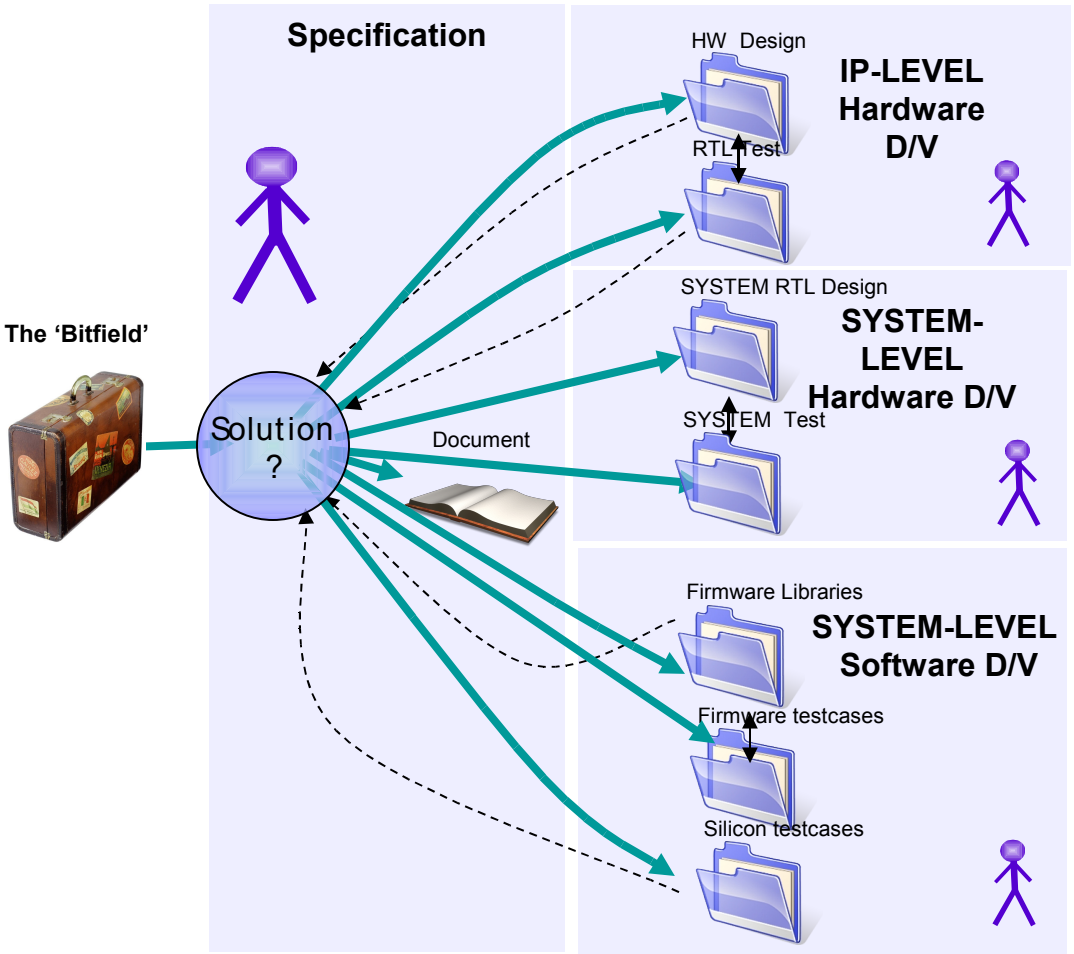
ARM | 2008 Developers' Conference

Compressed Design Cycle



- **Features**
  - TTM has compressed the integration cycle
  - Earlier deliverables required
  - Parallel/Concurrent process, not Waterfall
- **Problems**
  - No refinement or milestones
  - Very difficult to keep specification synchronised
  - High-Level of interdependency means high cost of incremental changes and schedule slippages
  - Massive translation leading to designed-in errors and high resource costs
  - No central coherency meaning bugs are caught farther downstream
  - Doesn't scale to increasing complexity
- **This applies to each suitcase (Bitfield)**
  - 45 flights of 220 people with 3 suitcases taking off at the same time = how many lost and missing ??

High-level Requirements - How do we solve?



- **Single Source** solution
- Efficient modelling of bitfields, registers, systems across the different design disciplines to allow **collaboration**
- **Auto-generation** of multiple output views
- **Standardization** of data across the company
- **Increased Quality** through Correct-By-Construction methodology
- Promote **reusability and modularisation**
- Ensure **fast generation** and ability to regress easily
- Ensure system **scalability** e.g. for 10,000s of registers
- Design to be **interoperable** via IP-XACT
- Aim to be **non-disruptive** to current flows
- **Easy to Use** environment across all major OS.

Register and memory map management

High-level Solution

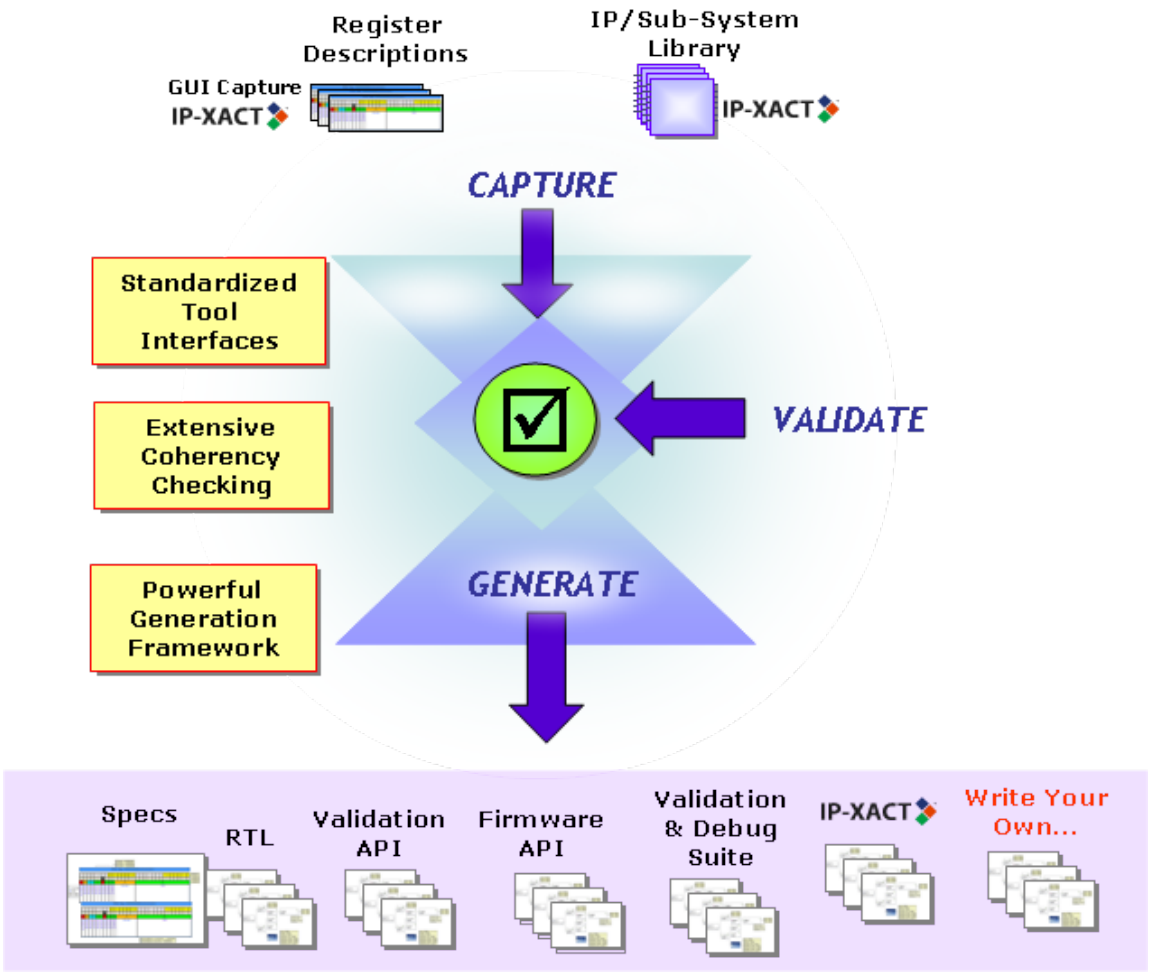
duolog technologies
www.duolog.com

ARM 2008
Developers' Conference

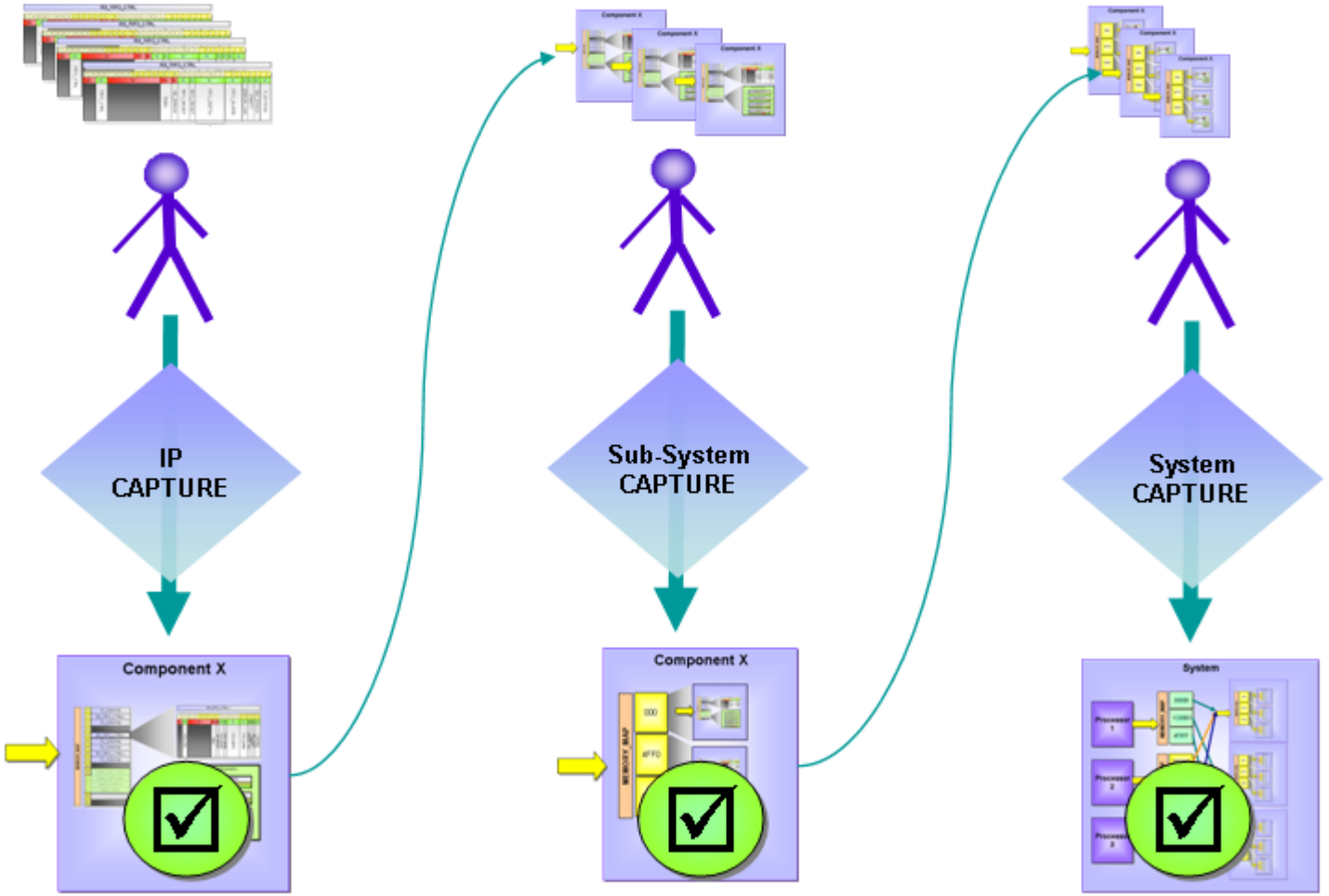
HW/SW Interface Management Solution

- Duolog have developed ‘BITWISE’ to manage HW/SW interface:
  - Efficient Model
  - User Friendly Design capture
  - IP and System Design flows
  - Standardized interfaces
  - Correct-by-Construction methodology
  - Flexible Generator Mechanism

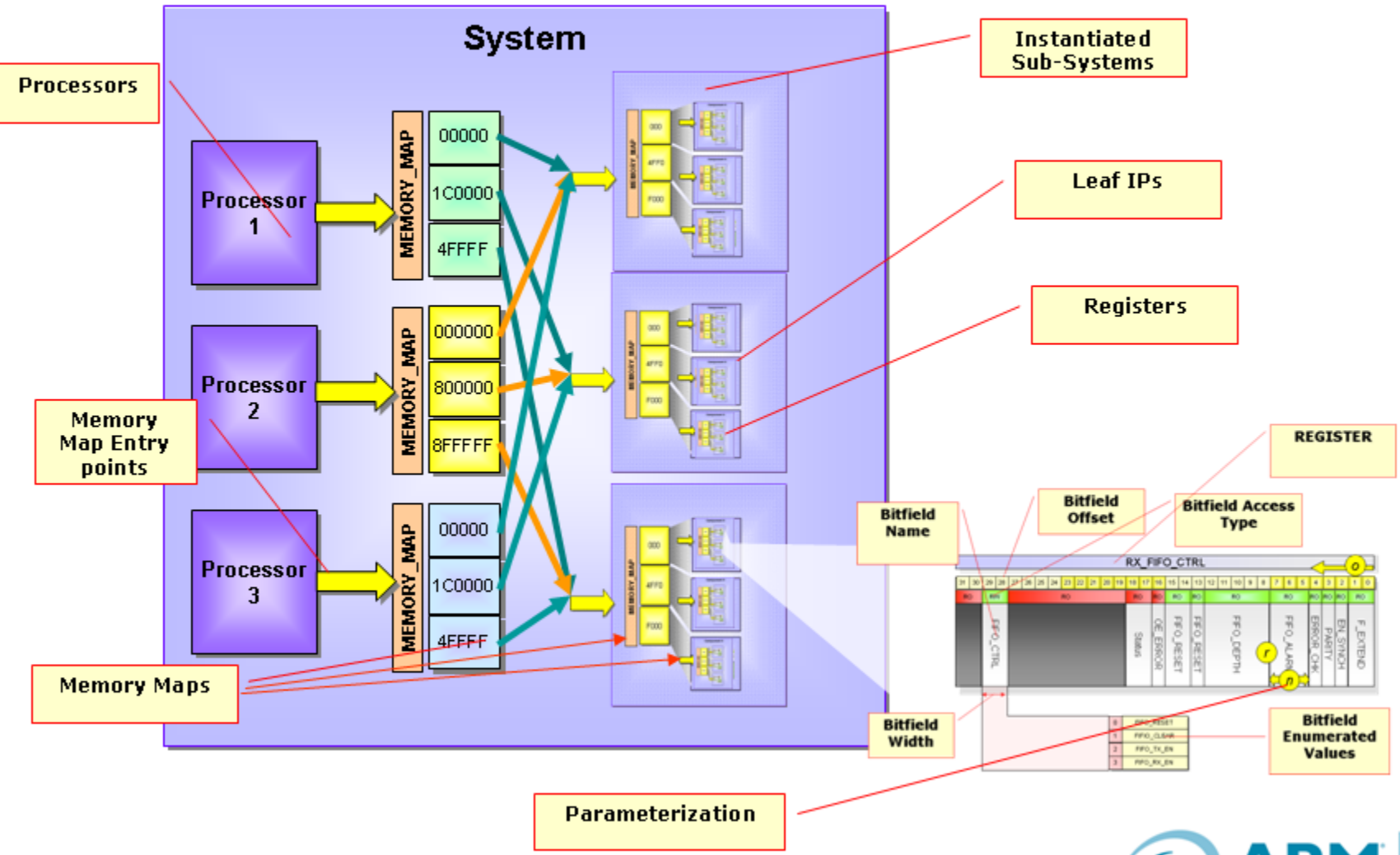
‘Bitwise’ Data Flow



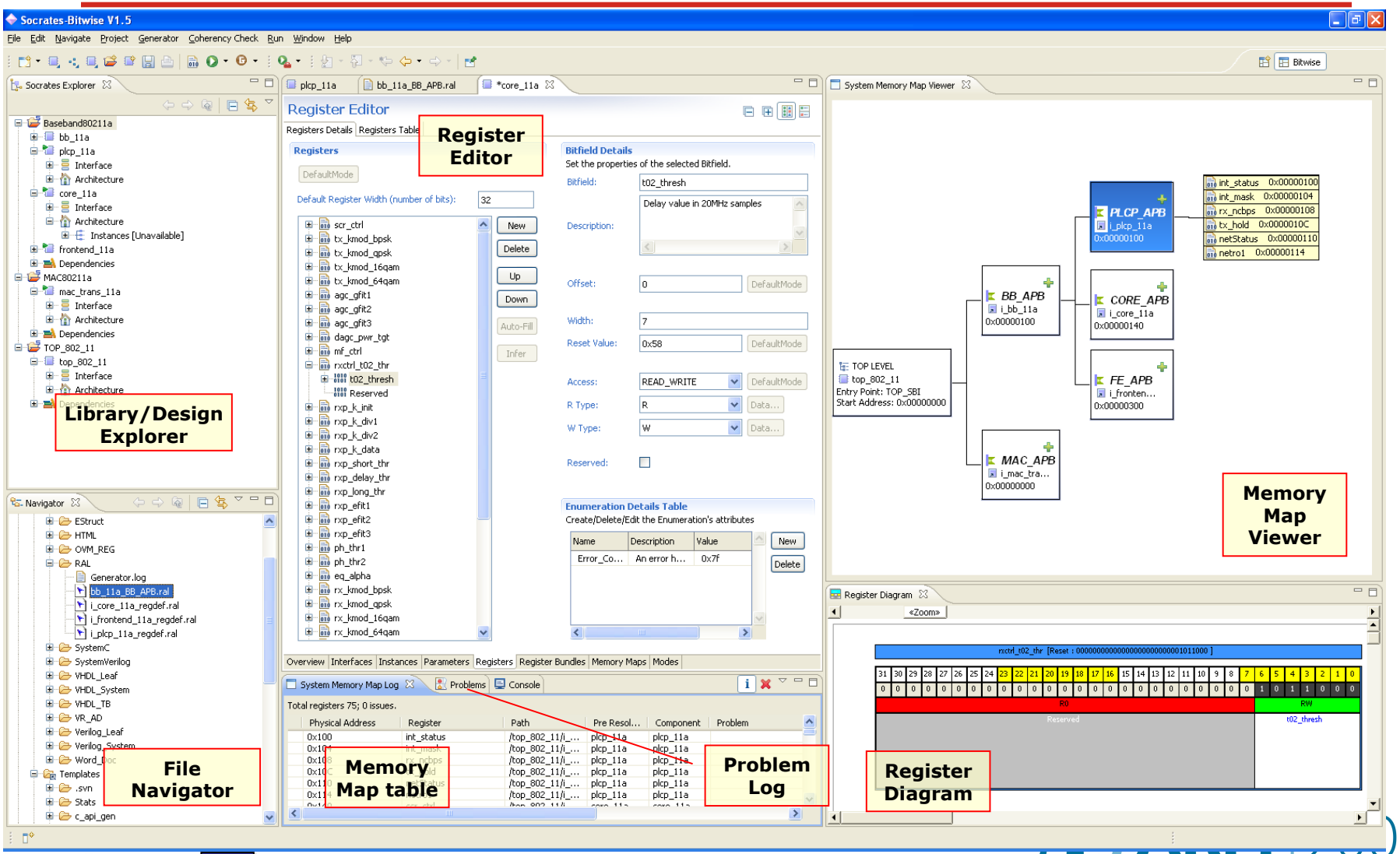
IP & System Design Flows



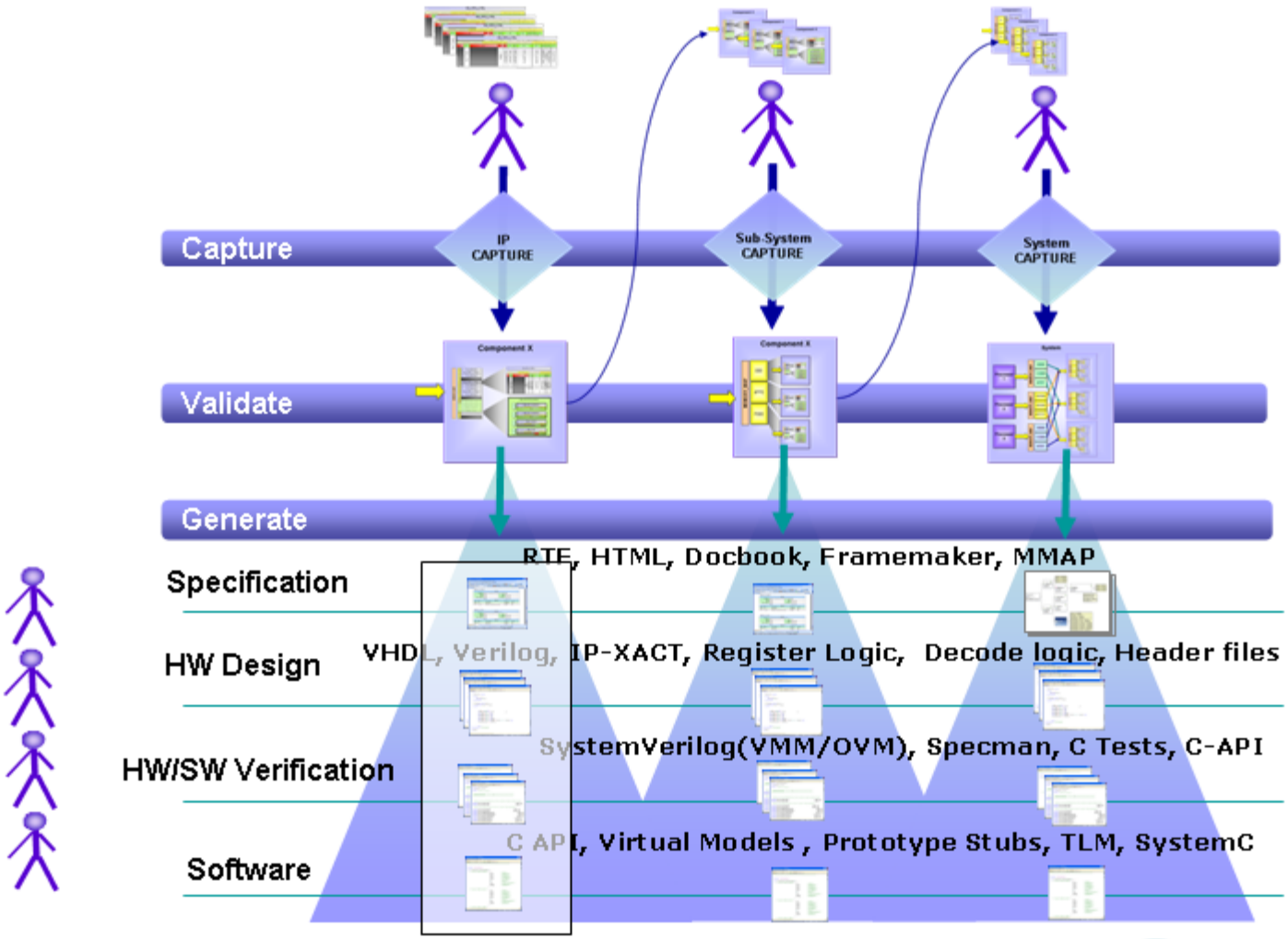
Modelling the System



User Friendly GUI



IP & System Design Flows



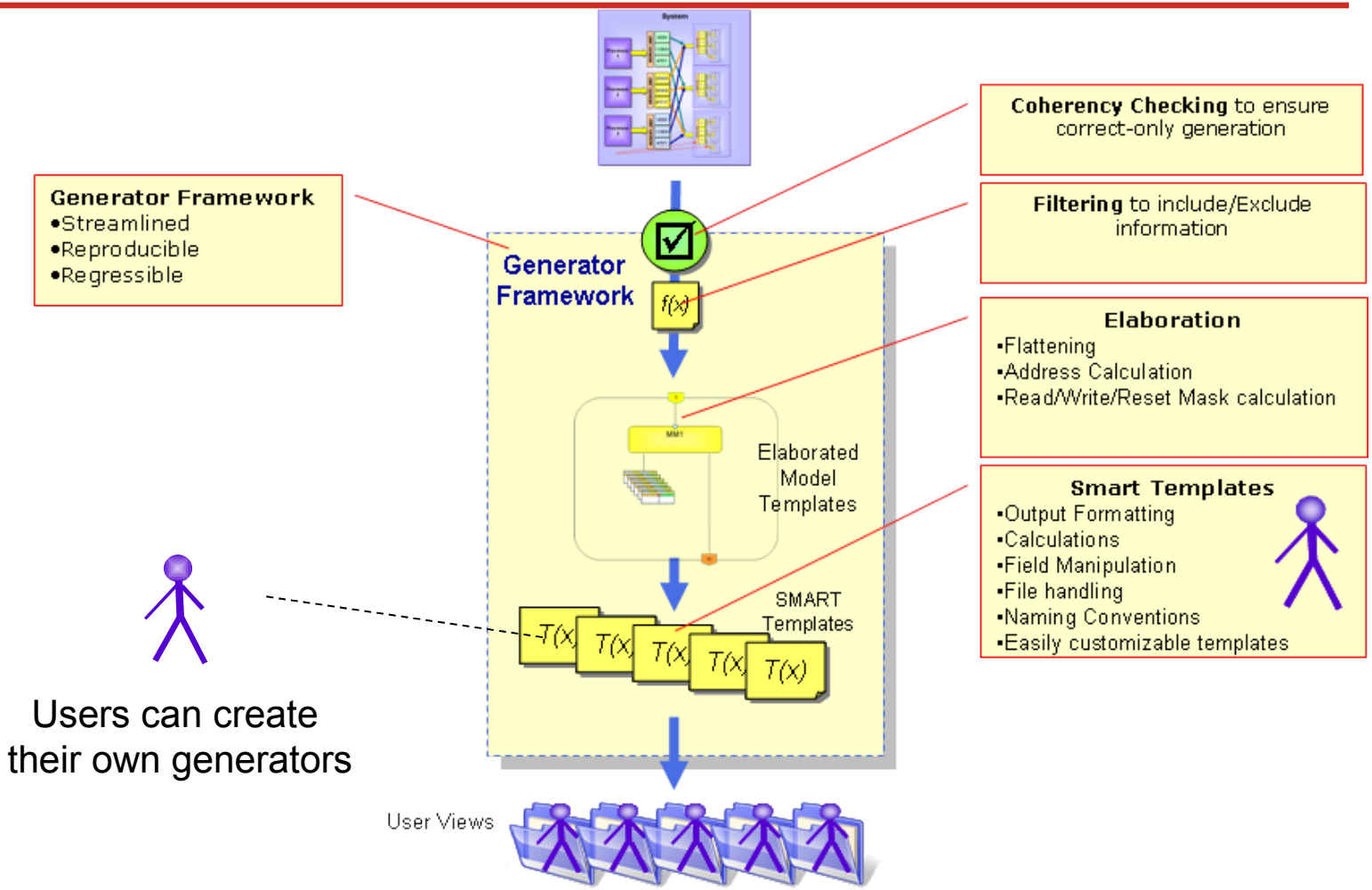
Standardized Interface with IP-XACT™

- Use IP-XACT™ as a standardized communication mechanism
- It is the best standard for register and memory map management
- Bitwise abstracts/decouples IP-XACT™ for user-friendliness, completeness and simplicity
- Import/Export IP-XACT™ 1.1, 1.2,1.4
  - Extract registers, bitfields, memory maps, etc
- Duolog has worked with TI and IP vendors to ensure alignment
- Main guideline - keep it simple

Correct-By-Construction

- Comprehensive ‘coherency checks’ to identify incorrect data
- Coherency checks run on-demand and on-the-fly as the user is entering data
- Coherency checks can be run on data imported to the system
- Successful coherency checking must be performed before any output data is generated

Generator Framework



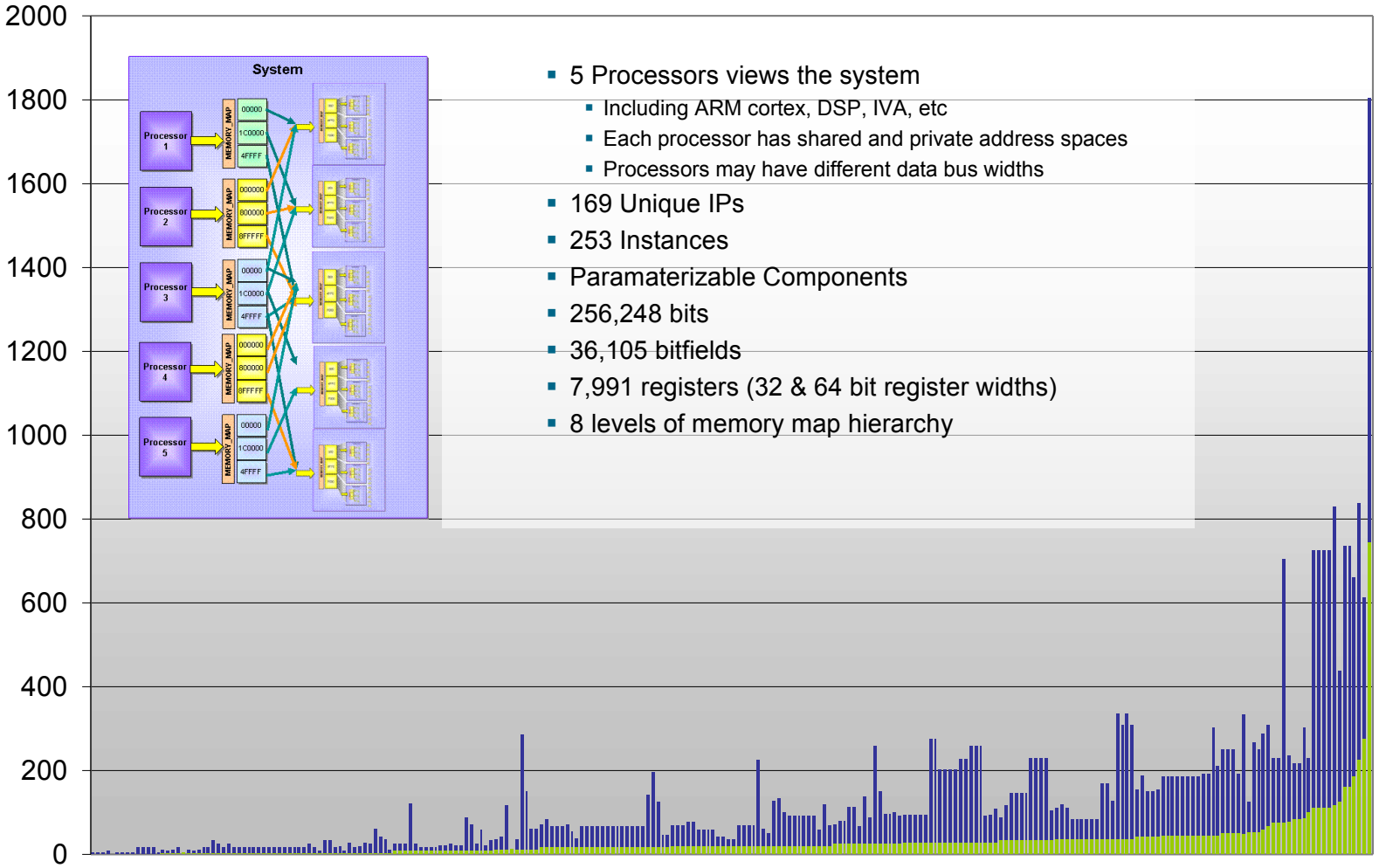
Case Study

Case study : Multimedia Processor

duolog technologies
www.duolog.com

ARM 2008
Developers' Conference

Case Study : SoC Application Processor

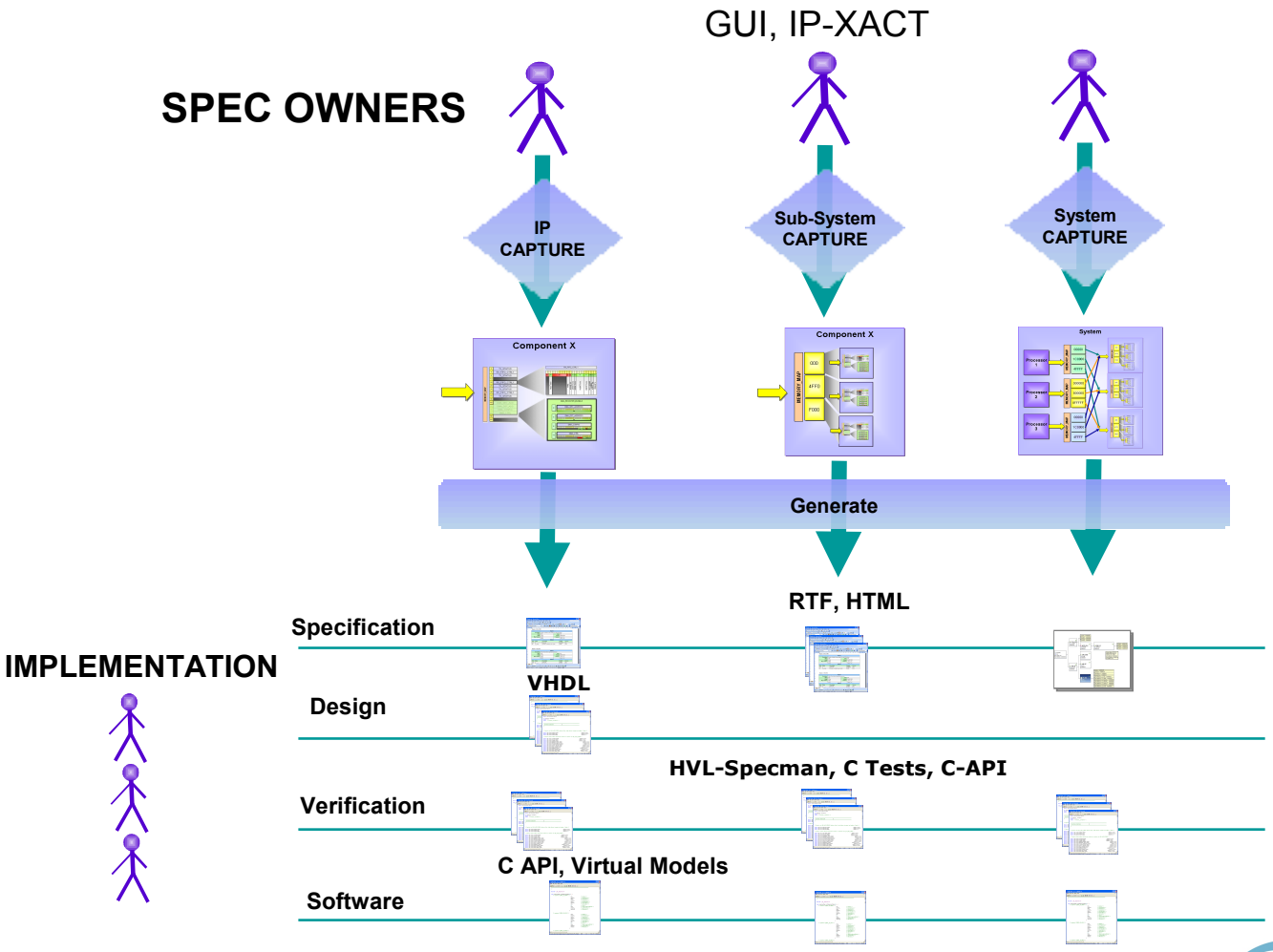


- 5 Processors views the system
  - Including ARM cortex, DSP, IVA, etc
  - Each processor has shared and private address spaces
  - Processors may have different data bus widths
- 169 Unique IPs
- 253 Instances
- Paramaterizable Components
- 256,248 bits
- 36,105 bitfields
- 7,991 registers (32 & 64 bit register widths)
- 8 levels of memory map hierarchy

duolog technologies

ARM | 2008 Developers' Conference

Soc Design Flow



Statistics

- Platform
  - 2.4GHz, 2Gbyte desktop machine with T3400 Intel dual core
  - Windows XP professional
- Outputs
  - 94 MBytes of Documentation (2 flavours)
  - 7.5 Million lines of C Code (3 flavours)
  - 1.26 Million lines of E code (2 flavours)
  - + Millions of lines of code of custom generators developed within Bitwise
- Generation Times
  - Coherency checking time = < 10 minutes
  - Generation time = average 5 minutes / generator

Benefits

- **Improved Quality**
  - Single-source specification in a single location – Write once, generate all
  - Perfect-by-construction methodology
  - Comprehensive coherency checking
- **Improved Productivity**
  - Consolidate information across multiple users
  - Promote early collaboration between engineering teams
  - Eliminate manual processes
    - Release engineering talent to perform more creative tasks
  - Single reference point and terminology for hardware & software teams
- **Reduced schedule**
  - Facilitate re-use methodology
  - Immediate turn-around from spec to generated views
  - Perfect-by-construction™ ➔ less debug, less re-work
  - CLI to run regressions
- **Reduced Costs**
  - 30x+ resource savings over manual based flows

Register and memory map management

Conclusion

duolog technologies
www.duolog.com

ARM 2008
Developers' Conference

Conclusion

- There are key pressures requiring early and robust HW/SW integration of ever more complex systems.
- A single-source specification can be utilised to provide a correct-by-construction methodology
- Current key focus is on auto-generation of verification collateral that ensures robust SW integration
- Software view of the system could be delivered much earlier in the design cycle before HW has been fully integrated
- A good solution can ensure fewer bugs @ less cost in less time.

Requirements Coverage

	IP Libraries	Full Register Model	High-level GUI	IP-XACT Import/Export	Eclipse	Coherency Check Framework	Generator Framework	CLI
Single Source solution	●	●						
Efficient capture of information across the different design disciplines : **Collaboration**		●	●					
Auto-Generation of multiple output views to keep information synchronous and up-to-date			●			●	●	●
Standardization of data across the company		●		●			●	
Increased Quality through rigorous specification DRC or Coherency checks	●					●		
Promote **reusability and modularisation**	●						●	
Ensure **fast generation** and ability to regress easily							●	●
Ensure system **scalability** e.g. for 10,000s of registers	●	●			●			
Design to be **interoperable**				●	●	●		
Aim to be **non-disruptive** to current flows				●	●		●	

Acknowledgements

- TI – WTBU Team in Nice
  - Hedi Boufaied
  - Bertrand Blanc
  - Bob Maaraoui
  - Denis Kuntz

Thank you – Questions?