

IP-XACT Usage Survey Results

Sept 2010

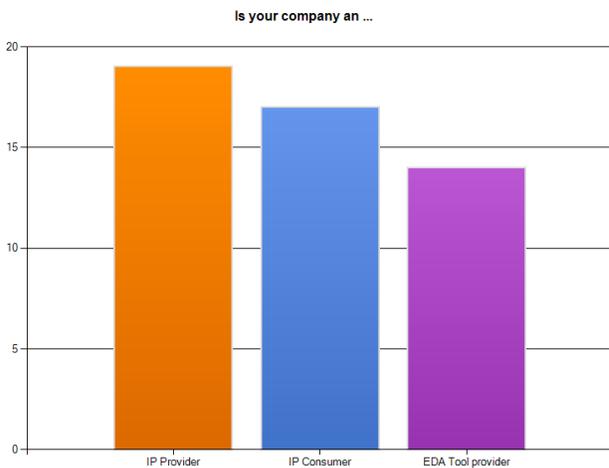
Compiled by

David Murray

David.murray@duolog.com

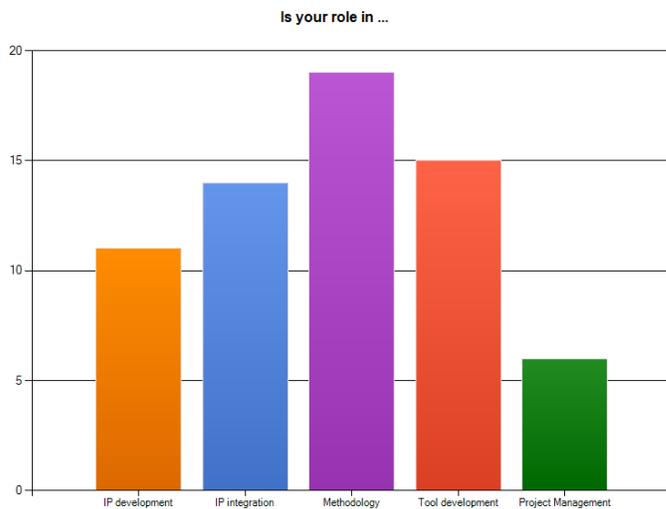
The survey had 41 responses representing 33 companies. The only mandatory question was email address in order to qualify the survey. Company/individual results and product references have been removed.

Who are you?



Other

- Consulting and training firm
- Semiconductor supplier
- Research Institute
- microchip manufacturer
- Mostly IP Provider
- Mobile Product based company
- Design for Reuse consultant

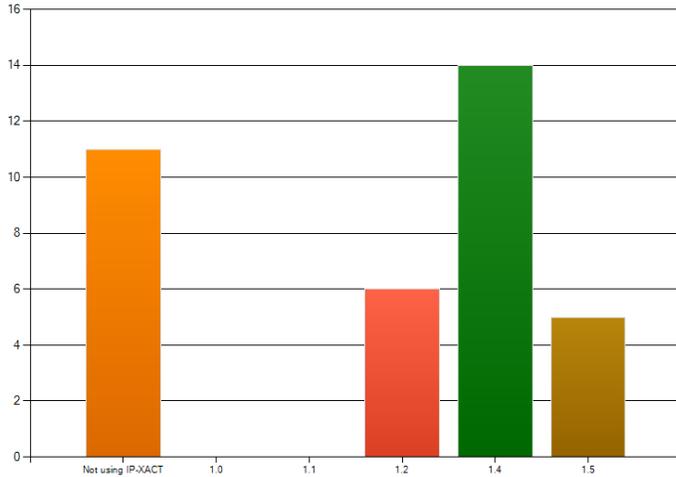


Other

- Consultant
- Executive
- Research
- Technical marketing
- Mostly IP Development and Methodology
- IP model integration
- Technical Training & Communications
- Executive Management
- SOC verification

IP-XACT versions

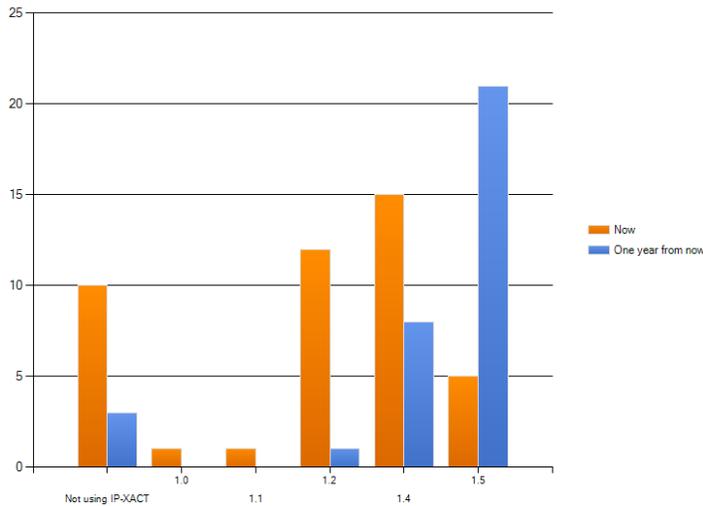
Which version of IP-XACT are you using the most?



Other

- *Proprietary XML Format*
- *...with [EDA tool]*
- *have our own schema*
- *all versions are needed*
- *All the above*

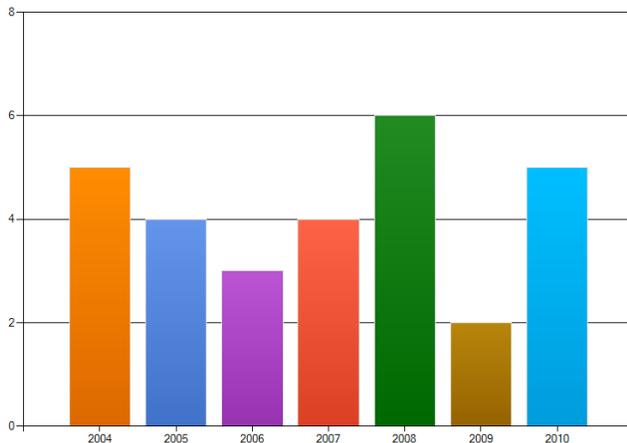
Which versions of IP-XACT are you using?



Other

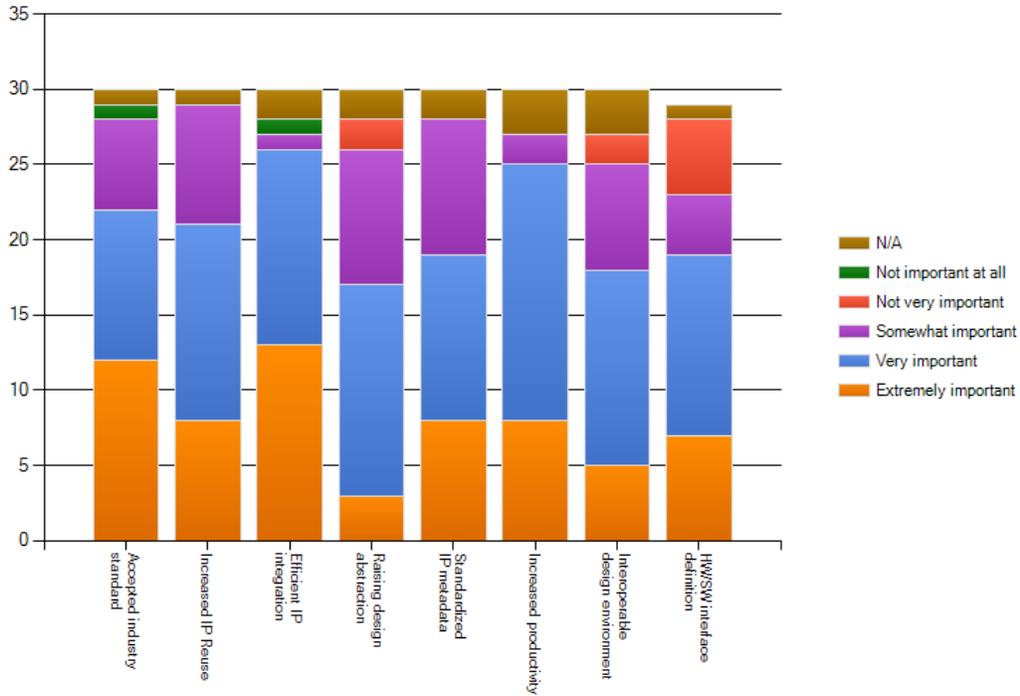
- *I'm not planning to use it but would like to influence the standard from a HW/FW interface point of view.*
- *Proprietary XML Format*
- *Hopefully later better version will be available.*

What year did you start using SPIRIT/IP-XACT?



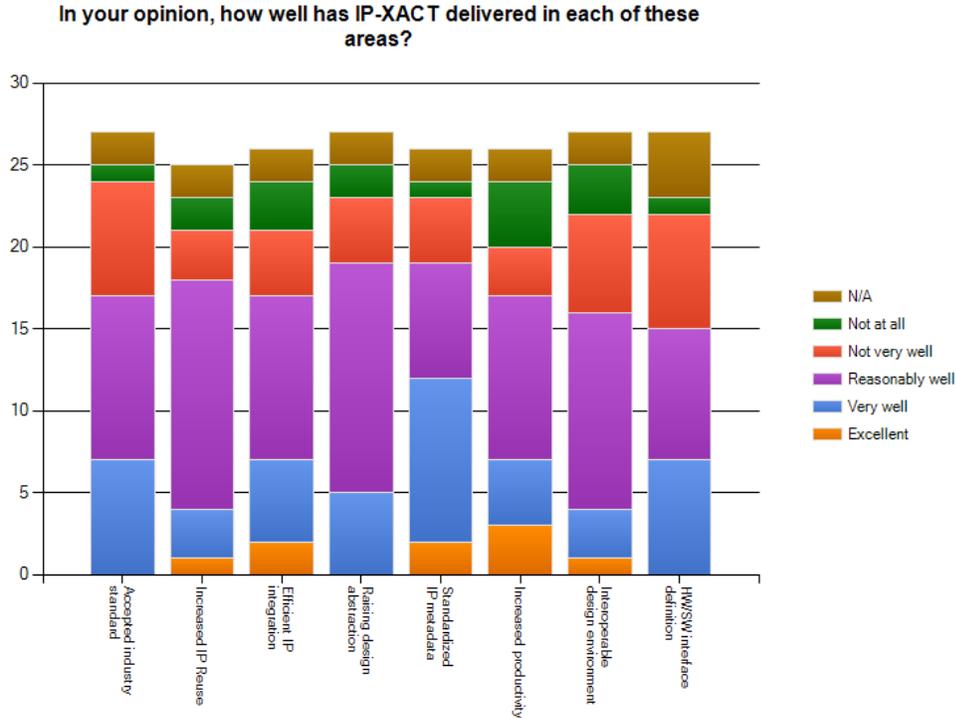
IP-XACT Adoption

What was the relative importance of each of the following in your decision to adopt IP-XACT?



Other

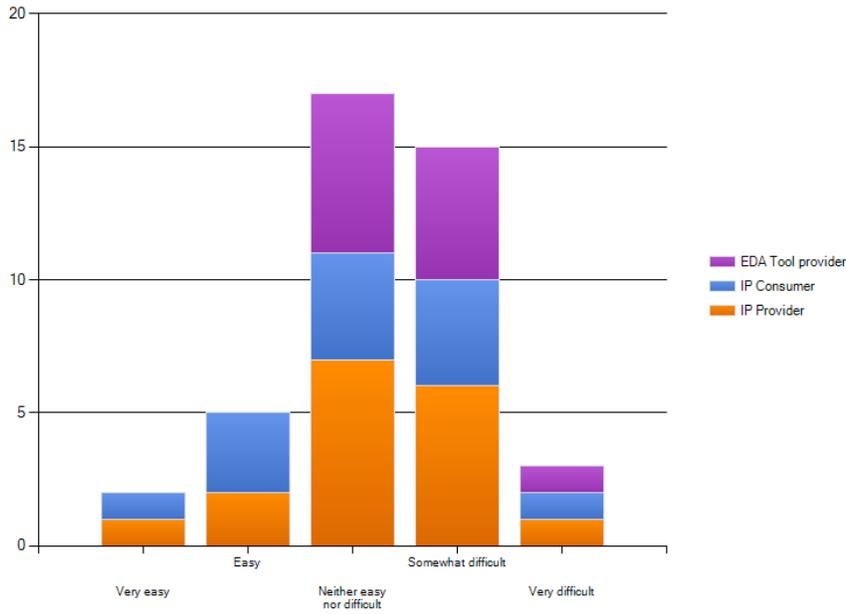
- *I have not adopted it but the HW/SW interface is very important, hence my interest.*
- *IP quality assessment*
- *Requested first by a partner and then by a customer for an individual core. Without the request to deliver IP-XACT descriptions, it would not have been created*
- *We haven't adopted IP-XACT because our investigations suggested that it was too heavily focused on the hardware and fell short at the software device driver / service level.*



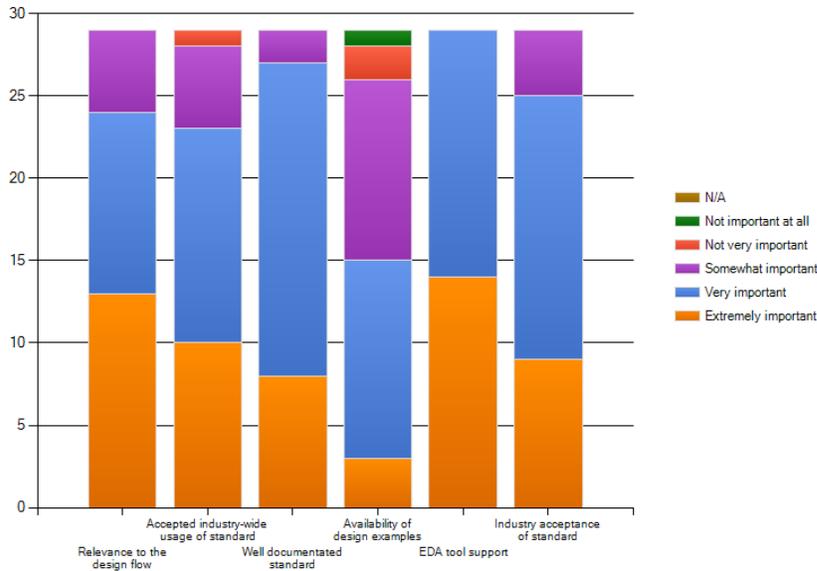
Other

- We are just looking into using IP-XACT
- I don't do design work so I have not used IP-XACT so I don't know how well it works.
- Efficiency got compromised in my experience because of bad timing circumstances: bus definition and IPXACT models of IP where not ready / mature by the time integration needed them
- I am still experimenting with it for internal use only.
- The problem seems to be that the IP packaging, provided by the IP vendors, is not as good as it could be.
- Have not seen real industrial demand for IP-XACT deliverables. Even the one customer deliverable may have been pro forma (to satisfy a checklist) than for real use.
- Standardizing on hardware specification is too short-sighted in our view. Each block of hardware also needs supporting drivers and services to be useful at a system level.
- I have never seen anyone actually using IP-XACT for a real chip design. It is improving. I used to get blank stares when I would ask my vendors for an IP-XACT file. Now they at least have heard of it so they will add it to the action items list.
- But so far I have never had anyone deliver an IP-XACT file.

How easy have you found it to adopt IP-XACT?



How important do you think are the following factors in facilitating the adoption of IP-XACT?



Other:

- *If users don't want to use it, they won't ask for it. If they don't ask for it, it won't be provided. Self-limiting feedback loop.*

Do you have any suggestions for improving IP-XACT adoption

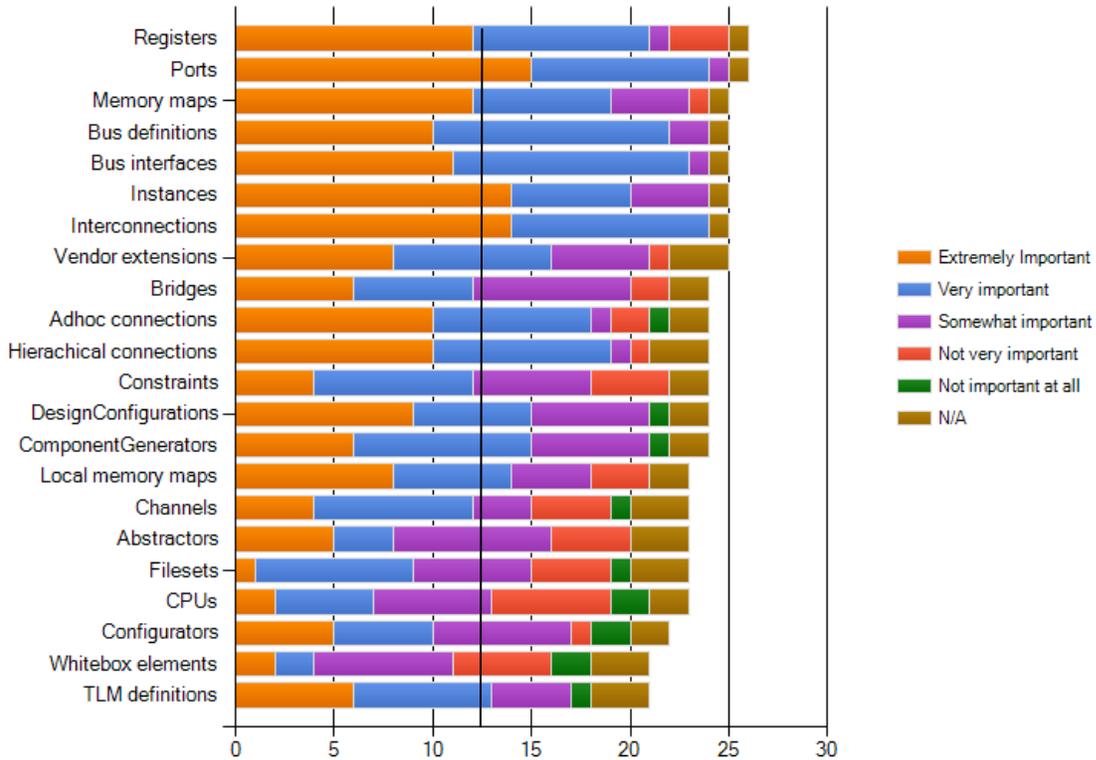
- More industry visibility perhaps roadmap for connection to physical IP data and tools
- More documentation on usage and adoption strategies would be helpful. I usually find xml readers or spec documentation and not a lot on how to leverage all this wonderful work.
- Consolidation in working groups are rather slow.
- Define a validation suite(s)
- The IP packaging itself should be standardized.
- IP-XACT and the associated tools should enable seamless replacement of TLM with RTL components. Moreover, IP-XACT should position itself in terms of support for different abstraction layers. What would be the benefit for System integrators and EDA companies if they would support IP-XACT in their tools in design flows.
How is IP-XACT positioned to the activities in Model-Based Design? Maybe IP-XACT should demonstrate its applicability in a joint Reference Technology Platform for Platform-based SoC design.
- Add mandatory verification data to allow consumers to quickly compare how IP was verified - example: variations of parameters and config registers
- Has to have some demonstrated advantage that is real in building designs. Right now, Verilog or VHDL seems to do as well.
- Make improvements suggested by EDA vendors.
- Agreed bus and abstraction definitions
 - IP-XACT based IP exchange
 - IP-XACT based generator exchange
- Simple, but not too-simple, use-case to demonstrate value
- Clearer documentation and examples. It often difficult to determine the appropriate use of elements of the IP-XACT standard.
- IP_XACT Vendor bus definition and port mapping to bus def won't make any sense if the IP meta data exchanged between IP vendor and IP consumer. This requires agreement between all IP providers. still this won't help unless the bus defs are integration friendly. Otherwise it is time wasting job for IP provider to capture all the mapping if it is not really helping integration team in making use of the bus def mapping.

I would recommend delivering IP with out bufdef mapping and let SOC integrator decide on how to map & what to map and connect using SOC integration tools.

- From my investigations, I have concluded that IP-XACT is a standard being pushed predominantly from the hardware end. This is too narrow. The use of interchangeable IP is not just dependent on the hardware. It is also dependant on software too. So IP-XACT needs to encompass a software integration model as well.
At [], we looked into adopting IP-XACT but decided to go it alone because although we offer IP, we recognize the importance of the software layers too. We have therefore defined our own software integration model that allows us to interchange both hardware and software to build systems.
- Release an open sourced tool kit that contains some real designs along with IP-XACT files and the ability to build a System on a chip from these parts.
- Less verbose

IP-XACT Features

How important are the following features of the IP-XACT specification to your design flow?

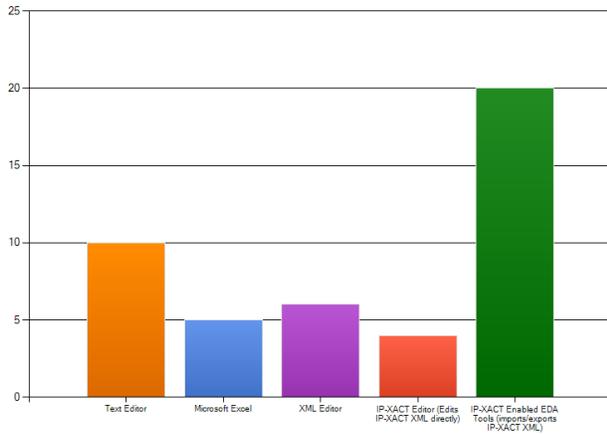


Other

- *Not sure whether we will use Whiteboxes or not*
- *Assumed TLM meant transactional ports?*
- *Not sure what configurators refers to*
- *Not familiar enough with the terms.*
- *All*

IP-XACT Interfacing

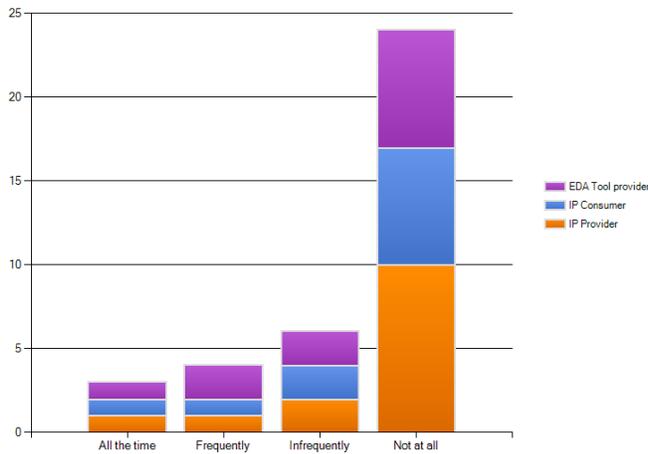
How do you capture IP-XACT data (please tick all that apply)?



Other

- Internal tools
- I've not used it.
- Proprietary Tool
- Only done twice. If demand had ever picked up there would be automatic generation.
- We currently don't use IP-XACT for reasons previously given.
- perl scripts

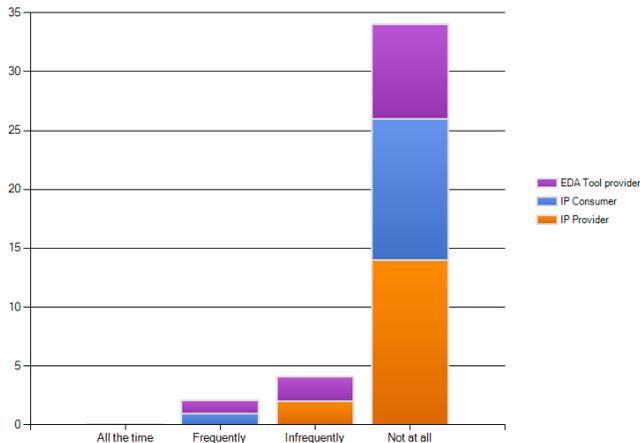
Do you use the Tight Generator Interface (TGI) to access IP-XACT data?



Other

- We expect to use TGI or vendor equivalents quite a bit
- I've not used it.
- I am unfamiliar with this interface.
- Own approach, similar to "all the time"
- may use in near future
- Will investigate
- Both TGI (SOAP) and TGI-like API (Java and Tcl)

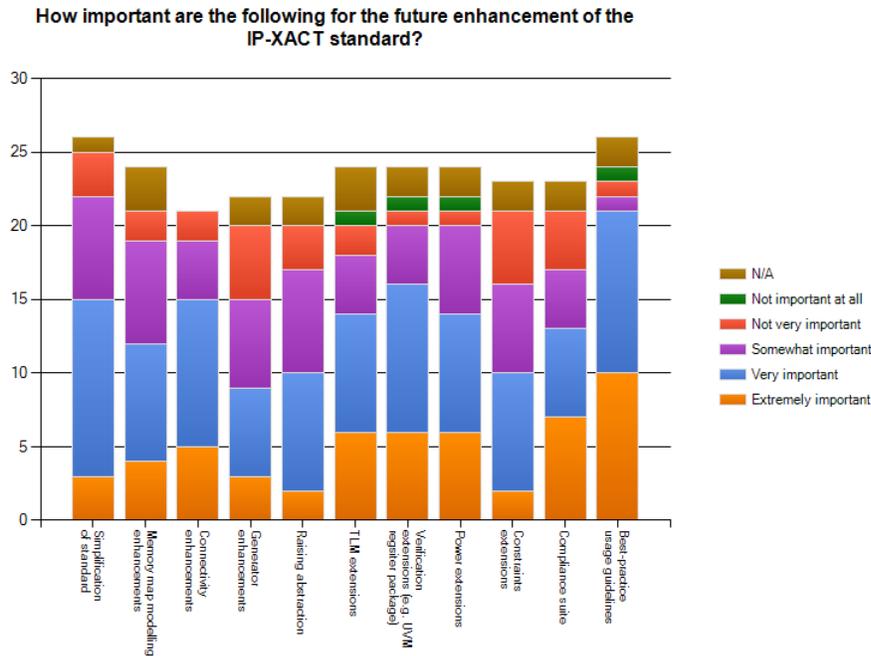
Do you use the SystemRDL language to describe register constructs?



Other

- Using internal solutions currently, looking into vendor register solutions
- I've not used it.
- may use in near future

IP-XACT Future Enhancements



Other

- *Not currently sure what proposed enhancements are out there, some may be more important than I have indicated.*
 - *The TGI needs to be extended with a full set of "new", "destroy", and "set" functions - You should be able to completely build and modify the data model from the TGI.*
 - *A general model for encapsulating physical data would be useful - these may simply be standard ways to reference external files (say CPF/UPF or P&R constraint file) - IP-XACT is close to being a general object packager for IP, but must have hooks for all IP data to be either included or referenced.*
- *I don't know yet what's really behind that.*
- *It should not evolve until there is a clear use model and clear user demand.*
- *Other*
 - (1) *Interface parameters*
 - (2) *Automatic interfaces*
 - (3) *Two-dimensional ports*

Any Final Feedback?

- I'm coming at it from the FW (SW) side. Unfortunately I have been too busy to learn IP-XACT to see how well it supports good HW/FW interface design but I suspect, based on what I have learned thus far, that it can be improved. I suspect that most that are involved in the HW/FW interface aspects of IP-XACT are HW engineers who, in my experience, don't fully understand the FW domain that they are designing an interface for. We should help HW engineers who use IP-XACT design HW that will integrate well with FW.
- Please proceed fast with the standardization!
Build a community which develops API's for different design platforms to create easily custom generators.
- The COMPLEX (<http://complex.offis.de>) IP project consortium would be interested of giving feed-back to the IP-XACT standardization activities.
Possible contributions:
 - MDA Model Driven Architecture (PDM Platform Dependant Model) to IP-XACT generation
 - TLM architecture description using IP-XACT
 - IP component (black and white-box) power models and possible ideas for the integration with IP-XACT.
- Will you be sharing the results of the survey in the future? If so, please contact me through LinkedIn or by the email address supplied earlier.
- Survey was too long.....
- No thanks.
- As previously mentioned, the focus of standardization should not be on the hardware alone. For true IP interoperability to be useful, IP must ship with software drivers / services etc that allow it to integrated quickly at the system level.
The creation of products is not a hardware only or software only activity. These domains must all work in with one another. Until IP-XACT can handle software integration, it will cease to be relevant to us.
Our company has been promoting a unified data model for some time now. Put simply, it means that designers from hardware, programmable hardware, and software domains can all work together within a single design platform. Hardware doesn't exist in isolation and neither does software. So all IP that we create and ship with our products also has complete software driver / service level support as well.
I'd be happy to talk to you some more about it if you'd like to understand more about our approach.
- I have great hopes that IP-XACT can solve many of the IC design issues that we now face but so far it has not been widely accepted. It is time to start "Evangelizing" and show designers how it can really help them.
- At least the new additions to the standard should be less verbose.

Would you be interested in learning more about the activities of the IP-XACT standardization body?

