



**SDR Forum Design Process and Tools Working Group  
RFI Final Report**

**SDRF-06-P-0006-V1.0.0**

**(Formerly Approved Document SDRF-06-A-0006-V0.00)**

Approved  
September 2006

**TABLE OF CONTENTS**

<b>1</b>	<b>INTRODUCTION</b>	<b>4</b>
<b>2</b>	<b>RFI OVERVIEW</b>	<b>4</b>
<b>3</b>	<b>REPORTING FORMAT</b>	<b>6</b>
<b>4</b>	<b>DEMOGRAPHICS</b>	<b>6</b>
<b>5</b>	<b>CORE TECHNOLOGY</b>	<b>10</b>
<b>5.1</b>	<b>Demographics</b>	<b>10</b>
5.1.1	Technology Providers	10
5.1.2	Tool Providers	10
<b>5.2</b>	<b>Design Processes Utilized by Technology Developers</b>	<b>11</b>
<b>5.3</b>	<b>Design Processes Supported by Tools Vendors</b>	<b>11</b>
<b>5.4</b>	<b>Key Issues</b>	<b>12</b>
<b>6</b>	<b>SDR SYSTEMS</b>	<b>13</b>
<b>6.1</b>	<b>Demographics</b>	<b>13</b>
6.1.1	Technology Providers	13
6.1.2	Tools Providers	13
<b>6.2</b>	<b>Design Processes Utilized by Technology Developers</b>	<b>14</b>
<b>6.3</b>	<b>Design Processes Supported by Tools Vendors</b>	<b>15</b>
<b>6.4</b>	<b>Key Issues</b>	<b>16</b>
6.4.1	Technology Developer Key Issues	16
6.4.2	Tool Developer Key Issues	16
<b>7</b>	<b>WAVEFORM DEVELOPMENT</b>	<b>17</b>
<b>7.1</b>	<b>Demographics</b>	<b>17</b>
<b>7.2</b>	<b>Design Processes Utilized by Technology Developers</b>	<b>17</b>
<b>7.3</b>	<b>Design Processes Supported by Tools Vendors</b>	<b>19</b>
<b>7.4</b>	<b>Key Issues and Future Directions</b>	<b>19</b>
<b>8</b>	<b>SDR PLATFORM HARDWARE DEVELOPMENT</b>	<b>20</b>

<b>8.1</b>	<b>Demographics</b>	<b>21</b>
8.1.1	Technology Providers	21
8.1.2	Tool Providers	21
<b>8.2</b>	<b>Design Processes Utilized by Technology Developers</b>	<b>21</b>
<b>8.3</b>	<b>Design Processes Supported by Tools Vendors</b>	<b>22</b>
<b>8.4</b>	<b>Key Issues</b>	<b>22</b>
<b>9</b>	<b>SDR PLATFORM INFRASTRUCTURE DEVELOPMENT</b>	<b>23</b>
<b>9.1</b>	<b>Demographics</b>	<b>23</b>
9.1.1	Technology Providers	23
9.1.2	Tool Providers	23
<b>9.2</b>	<b>Design Processes Utilized by Technology Developers</b>	<b>23</b>
<b>9.3</b>	<b>Design Processes Supported by Tools Vendors</b>	<b>23</b>
<b>9.4</b>	<b>Key Issues</b>	<b>23</b>
<b>10</b>	<b>RFI SUMMARY</b>	<b>24</b>
<b>11</b>	<b>NEXT STEPS</b>	<b>25</b>
	<b>APPENDIX 1: DEFINITIONS</b>	<b>26</b>
	<b>APPENDIX 2: RAW RFI DATA</b>	<b>28</b>

## 1 Introduction

In June 2005, The SDR Forum Design Process and Tools Working Group (DPT-WG) published a Request for Information (RFI) seeking information regarding current design processes and tools used by industry in the creation of radio systems, hardware, software, waveforms, and components. The objective of the DPT-WG RFI was to identify best practices for design flows and tools used by industry to improve productivity and quality in the development and deployment of software defined radio technology such as might be utilized in the US Joint Tactical Radio System (JTRS) or the European End-to-End Reconfigurability (E2R) programs. In defining best practices, the DPT-WG intend to make recommendations in four key areas:

- Terminology: Development of a common terminology that can be used to describe the design flows and tools.
- Technology Push: Identify reference design flows that are utilized by industry in the development of software defined radio technology given the current state of the art in tools.
- Market Pull: Identify design flows that are desired by industry in the development of software defined radio technology, and the classes of tools necessary to facilitate each state within those design flows.
- Gaps: Identify the gaps that exist between technology push and market pull, and make recommendations to industry toward closing those gaps.

Responses to the RFI were received in July and August 2005, and the Design Process and Tools Working Group analyzed the responses in the following months. This document presents this analysis and makes recommendations on next steps.

## 2 RFI Overview

This DPT-WG RFI was intended for SDR technology providers and end users, as well as those who specify, design, and build software defined radios. The RFI was written to allow responses from a broad range of disciplines. Respondents could specify, manage, develop, test, document, or market software defined radios, or play any of these roles in a company or institution that supports software defined radio development. The RFI was presented to respondents in the form of an online “survey” hosted by Zoomerang. An MS Word version was also available.

The RFI began with a set of demographic questions identifying the responder and the responder’s affiliation. Identification was encouraged to allow the Working Group to make contact with respondents if more information or further clarification were required. However, anonymous responses were accepted.

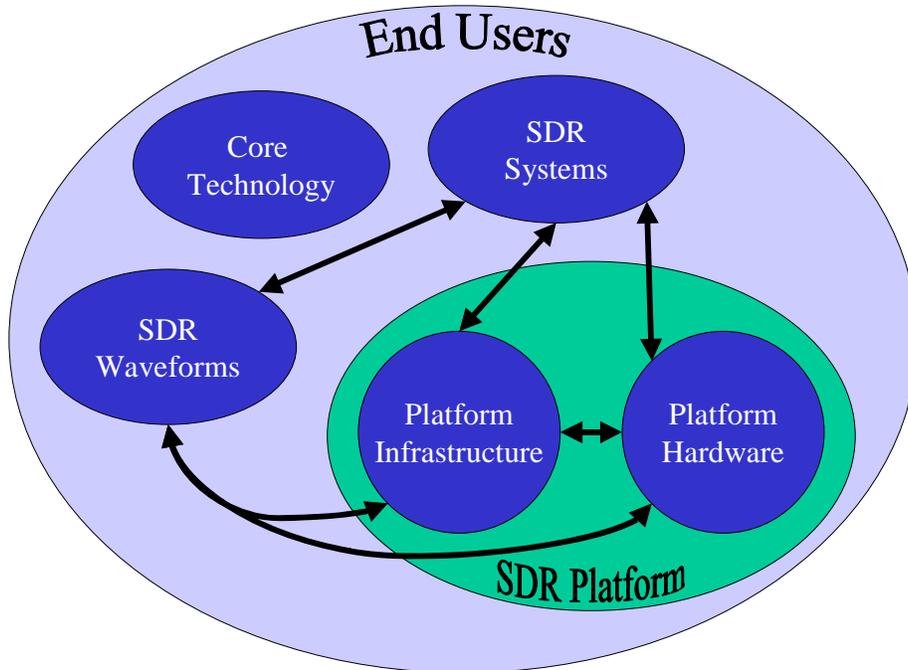
The RFI was separated into two main sections: one designated Software Radio Technology Providers and one designated Tool Developers. The section for Software Radio Technology Providers solicited information from people actively delivering

software radios to market as opposed to those who develop tools that enable the creation of software defined radios. This section gathered information that describes what actually goes on in SDR development with the goal of helping the DPT-WG understand which SDR development tasks and flows are well understood and well supported, and where users see a need for more assistance.

The Tools Developers Section solicited information from tool developers on the technology and solution that their tool provides. This section identified existing industry-provided tools that target specific parts of the software defined radio technology development flow, with the goal of helping the DPT-WG to identify and recommend reference design flows that can be utilized by industry in the development of software defined radio technology given the current state-of-the-art tools. Comparison between the two sections was used to identify any existing gaps in the design process that need to be addressed.

Each of the major sections of the RFI was divided into the five subsections, as illustrated in Figure 1:

- **Core Technology:** This subsection explored the creation of technology that is provided to radio manufacturers by third parties. Examples include components and other intellectual property (IP), standards, waveform specifications, tools, institutional research and development (R&D), etc.
- **Software Defined Radio Systems:** This subsection explored the creation of SDR technology at a system-wide level. For the purposes of the RFI, the building blocks for SDR systems are the software and hardware components created by other teams.
- **SDR Waveforms:** This subsection explored the creation of all or part of the waveform functionality.
- **SDR Platform Hardware:** This subsection explored the creation of all or part of the hardware (RF, mixed signal, digital) platform on which the waveform applications run.
- **SDR Platform Infrastructure:** This subsection explored the creation of the application-independent software/firmware infrastructure (device drivers/hardware abstraction, application frameworks, middleware, waveform services, etc.) that supports the SDR platform.



**Figure 1: RFI technology subsections and their associated dependencies**

### 3 Reporting Format

The reporting format utilized in this document is as follows:

1. Design processes will be presented as state diagrams. Feed forward paths are shown in black, feedback paths are shown in blue
2. Facts will be presented as normal text. Any theories or analyses that are presented based on these facts will appear in *italics*

The state diagrams presented in each section are largely the result of analysis and as such are not marked separately.

### 4 Demographics

The following tables provide an overview of the demographic information that was captured in the RFI. They are presented without analysis. Details with respect to specific categories are provided in the corresponding subsection. There were a total of 70 individual responses. Only 49 were used as data input for this RFI, the remaining responses were incomplete and therefore insufficient for inclusion.

3.	What is your product domain?	Number of Responses	Response Ratio
	Military 	24	41%
	Commercial 	15	25%
	Public Safety 	5	8%
	Other, Please Specify 	15	25%
	<b>Total</b>	59	100%

4.	What is your company's primary function?	Number of Responses	Response Ratio
	SDR technology developer – radiosets provider 	30	51%
	SDR technology end user – specify and purchase radiosets	0	0%
	IP Provider	2	3%
	Tool Vendor 	10	17%
	Other, Please Specify 	17	29%
	<b>Total</b>	59	100%

5.	Size of company (number of employees):	Number of Responses	Response Ratio
	< 50 	10	17%
	< 500 	14	24%
	< or = 5000 	11	19%
	> 5000 	23	40%
	<b>Total</b>	58	100%

6.	Which category best describes the SDR applications that you are involved with?	Number of Responses	Response Ratio
----	--	---------------------	----------------

Tactical Military Communications		18	32%
Communications Infrastructure		2	4%
Personal LAN		1	2%
Fixed Broadband Wireless Access		3	5%
Satellite Communications Radar		3	5%
Commercial Cellular		4	7%
Image Acquisition/Analysis		0	0%
Video Transmission/Compression		0	0%
Audio Transmission		0	0%
Networking/Telecommunications (data protocols)		3	5%
All of the above		11	19%
Other, Please Specify		12	21%
<b>Total</b>		57	100%

7.	Which describes your company's primary market segment?	Number of Responses	Response Ratio	
	Automotive	1	2%	
	Computer	0	0%	
	Consumer	1	2%	
	Data Communications		3	5%
	Medical	0	0%	
	Military/Aerospace		27	49%
	Semiconductor		4	7%
	Telecommunications		11	20%
	Electronic Design Automation	1	2%	
	Component Supplier	0	0%	
	Other, Please Specify		7	13%
	<b>Total</b>	55	100%	

8.	What types of SDR development problems does your company/department address (you can choose more than one)?	Number of Responses	Response Ratio
----	---	---------------------	----------------

The development of a new SDR platform to support one or more new waveforms		24	67%
The development of a new waveform to operate on an existing SDR platform		16	44%
The development of a new SDR platform to support one or more existing waveforms		23	64%
The development of a feature upgrade to an existing waveform operating on an existing SDR platform		11	31%
The development of a performance upgrade to an existing SDR platform		11	31%
The porting of an existing waveform to an existing SDR platform that may incorporate a different architecture than the waveform was originally designed for		13	36%
Interoperation of software defined radios in a system of systems		15	42%
Other, Please Specify		6	17%

**9. What types of SDR development problem does your tool help to address (you can choose more than one)?**

		<b>Number of Responses</b>	<b>Response Ratio</b>
The development of a new SDR platform to support one or more new waveforms		12	80%
The development of a new waveform to operate on an existing SDR platform		8	53%
The development of a new SDR platform to support one or more existing waveforms		9	60%
The development of a feature upgrade to an existing waveform operating on an existing SDR platform		5	33%
The development of a performance upgrade to an existing SDR platform		5	33%
The porting of an existing waveform to an existing SDR platform that may incorporate a different architecture than the waveform was originally designed for		5	33%
Interoperation of software defined radios in a system of systems		4	27%
Other, Please Specify		2	13%

## 5 Core Technology

The Core Technology section of the DPT-WG RFI explored the design processes associated with the creation of technologies that are supplied to radio manufacturers to be included in their radio systems, and the tools that support these design processes. Examples of core technology include components and other intellectual property, standards, waveform specifications, tools, institutional R&D, etc. The results of this section of the RFI follow.

### 5.1 Demographics

Of the seven usable responses to the core technology section, five were from core technology providers, the remaining two were from tool providers. Respondents came from various segments of the SDR community.

*However, the small sample size limits our confidence in the completeness of this data set.*

#### 5.1.1 Technology Providers

Of the five core technology providers, three considered themselves technology providers/radioset providers, two were IP providers. They were predominately commercial companies, with one respondent from the military (Mil/Aero) sector.

Two companies had an employee base of from 50 to 5,000 people; the remaining three companies were considerably smaller, at fewer than 50 people.

The SDR applications developed by the respondents include:

- Fixed broadband wireless access (two companies)
- Commercial cellular (one company)
- Satellite communications RADAR (one company)
- Other (one company)

The technology providers were split with respect to the manner in which tools are selected for use; three select tools on a project-by-project basis, and two abide by corporate decisions for tool selection.

Teams sizes were all less than 15 people, with one team of fewer than 5 people.

#### 5.1.2 Tool Providers

Two tool providers responded to the core technology section. One respondent was both a tool provider and a technology provider (semiconductors), but was accounted for in this survey as a tool provider. The two tool providers stated that the average team size that their tool supports were teams of fewer than fifteen persons. *This is likely an assessment of the customers' choice of team size. It is not a function or limitation of the tool.*

## 5.2 Design Processes Utilized by Technology Developers

In our analysis, the technology developers predominantly are generating specifications in text formats (*prose*) supplemented by diagrams. These are not executable specifications (such as would be used in model-based design), but rather human readable documents. Consequently, text editors are the most prevalent front-end tools for this group. From these specifications, the core technology provider is typically creating IP or standards, in the form of software for algorithms, simulation models, or more text documents for specifications. Few standards for specifications or IP are generated. Consequently, there is little common tooling or interoperability.

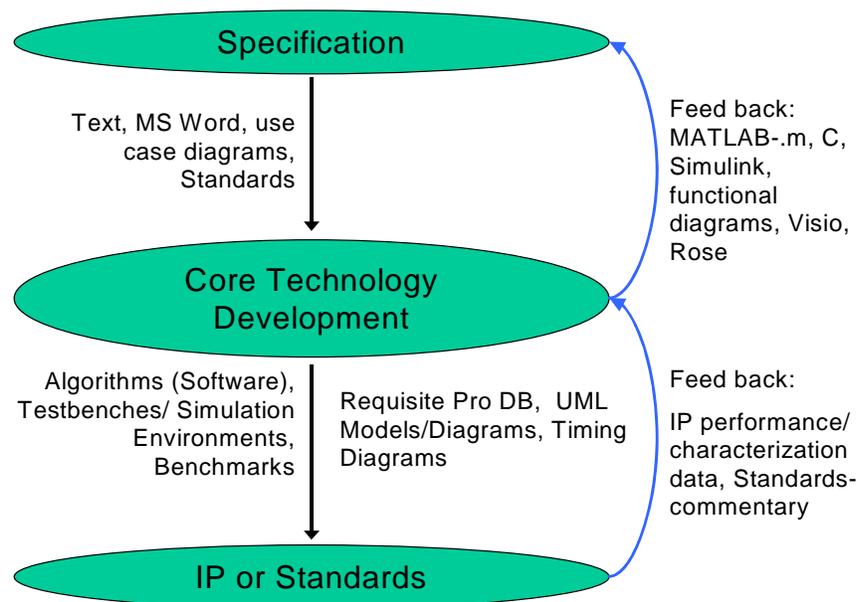
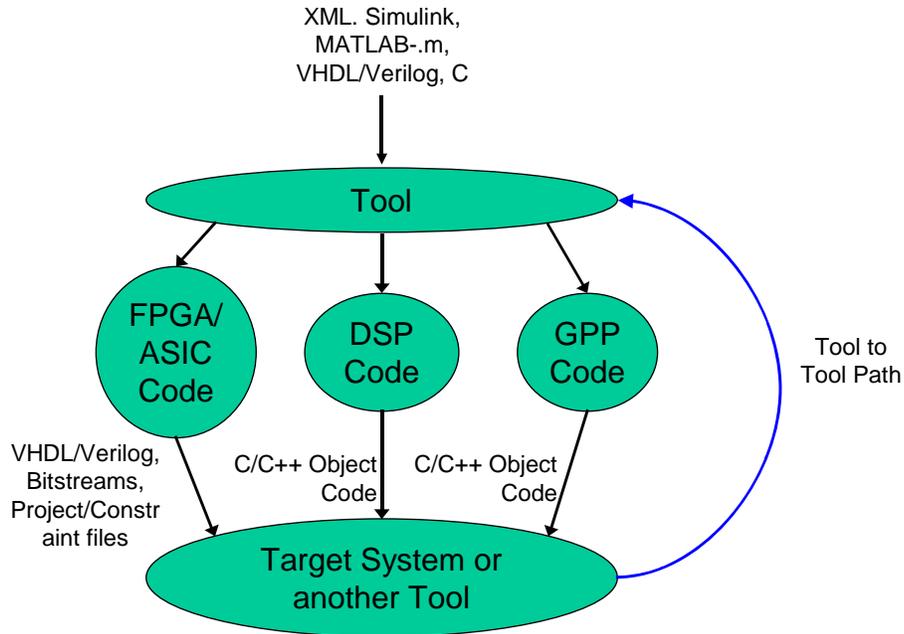


Figure 2: High-level design process used in the creation of core technology

## 5.3 Design Processes Supported by Tools Vendors

Tools vendors are generally supplying a point tool targeting a certain input format and/or a specific output target. Common input or specification formats are the eXtensible Markup Language (XML), The MathWorks' MATLAB® M and Simulink®, the VHSIC Hardware Design Language (VHDL) and Verilog Register Transfer Level (RTL), and the C programming language. The input specifications are used to create an implementation for a specific target, such as a general purpose processor (GPP), a digital signal processor (DSP), or “hardware” processors such as a Field Programmable Gate Array (FPGA) or an application-specific integrated circuit (ASIC). For example, RTL specifications are used for FPGA and ASIC implementation, and C is used primarily for DSP and GPP targets. Tools such as MATLAB and Simulink are in some instances a common environment for either ASIC/FPGA implementation or GPP/DSP development. A common deficiency is the lack of standards that exist in these domains. This restricts tool interoperability and specification portability across hardware targets.



**Figure 3: High-level design process supported by tools vendors in the creation of core technology**

## 5.4 Key Issues

*The SDR community is currently seeing a convergence of engineers from multiple disciplines working very closely together. Examples of such a coalition include the system designer, the algorithm or waveform developer, and the FPGA and ASIC hardware designer.*

*A lack of common languages, tools, and standards is seen as impeding productivity and effectiveness by this broad community.*

*For example:*

*There are no standards for interfaces between stages of a top-down design flow.*

*In addition, there are no standards for interoperability between tools, especially when they cross disciplines, such as the hardware/software interface “C” debuggers, RTL simulators, etc.*

## 6 SDR Systems

The SDR Systems section of the DPT-WG RFI explored the design processes associated with the creation of radios at a system-wide level, and the tools that support the creation of radios at this level. The building blocks for this process were defined as the software and hardware components created by other teams. In addition, this section of the RFI queried responses from individuals who market, document, manage, or otherwise work with radio products at an end-to-end level. The results of this section of the RFI are as follows.

### 6.1 Demographics

The SDR Forum Design Process and Tools Working Group received nine usable responses to the SDR systems development sections of the RFI: six from technology developers and three from tools vendors.

#### 6.1.1 Technology Providers

Of the six technology providers, three identified their primary domain as military, two as public safety and one as commercial. One company had fewer than 50 employees; the others all had more than 5000. Individual team sizes ranged from fewer than 5 to 150. Most respondents were developers, with one manager and one respondent responsible for specification and testing. One company has a dedicated tools specification group; one different company has an in-house tools development group. Tool selection was universally done on a project-by-project basis.

#### 6.1.2 Tools Providers

Of the three tools providers, two identified their primary focus as Military and one as R&D. One company had fewer than 50 employees; the other two had fewer than 500. All team sizes were between 5 and 50.

The SDR Systems tool vendors who responded to the survey addressed strongly overlapping areas. All tool offerings supported

- Model-based specification of SCA platforms and applications.
- Generation of XML descriptor files, packaging source code and documentation.
- Run-time monitoring of execution.

Additional reported tool capabilities included

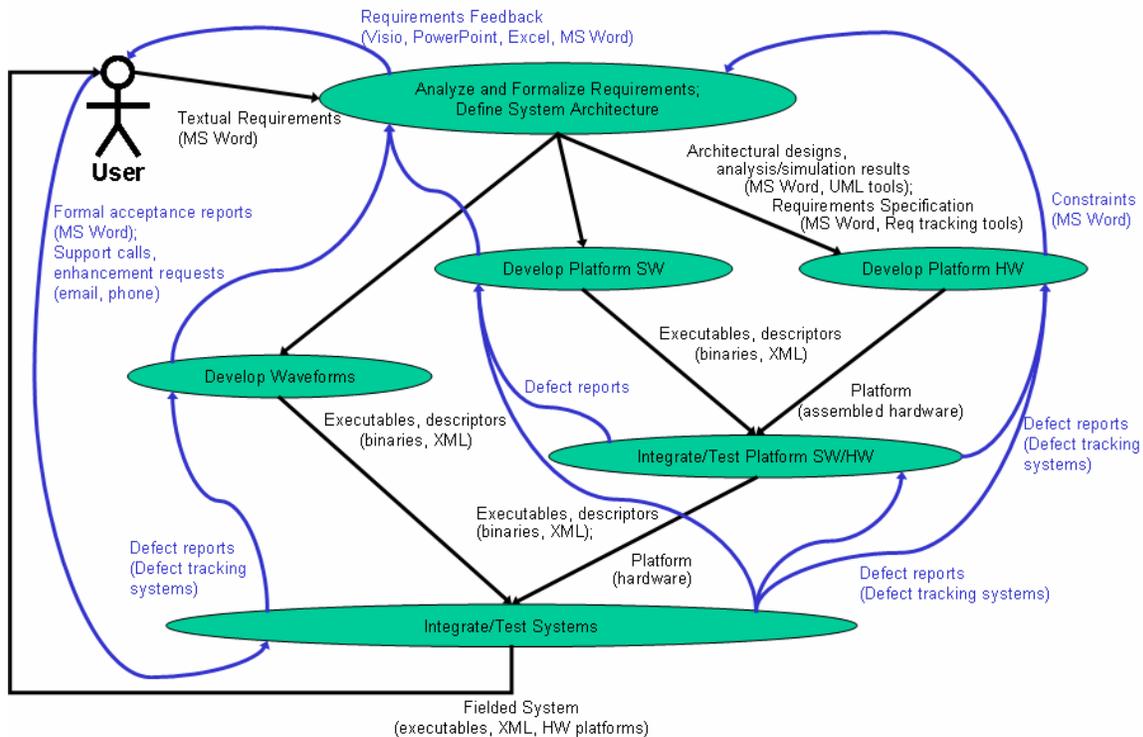
- Test driver generation
- Deployment specification and enforcement

Some vendors additionally had related non-tool offerings such as Software Communications Architecture (SCA) Core Frameworks and Object Request Brokers (ORBs).

*The breadth of SDR System Tool Developer survey respondents is limited. This should be borne in mind when interpreting responses.*

## 6.2 Design Processes Utilized by Technology Developers

Due to the limited number and detail of responses, the design processes used by technology developers and tools vendors have been combined into a single view.



**Figure 4: State diagram detailing the system development process**

The process shown in Figure 1 starts with the end user delivering textual system requirements in Microsoft Word format as input to a system requirements and architecture definition activity. Feedback on requirements is delivered in a range of Microsoft Office formats.

The outputs of the requirements formalization and architectural definition activity are inputs to the activities of developing platform hardware, platform software, and waveforms. Requirements are passed in Word and in proprietary requirements tracking tool formats. Designs and simulation results are passed as Word documents and also in Unified Modeling Language (UML) models. Feedback is returned as Word documents.

*This is the only place in the reported workflow where models appear as an interchange format. This is significant in light of interworking concerns expressed below. Users identify lack of tool interworking as a key issue and model interchange can enable this, so it is a best practice. However, communication through models was reported in only one response.*

The development activities deliver hardware, executables (as binary files) and descriptors (as XML files). Platform hardware and software are integrated in an integrate/test

platform activity. Defect reports in proprietary defect tracking system formats provide feedback to the platform development activities.

The output of the platform integration and waveform development activities are again hardware, executables, and descriptors. They are input to the system-level integration and test activity. Feedback is provided to the development activities and the platform integration, again as defect reports in proprietary formats.

The output of the system integration and test activity is a fielded system of executables, descriptors, and hardware. It closes the loop by going back to the user. The feedback flow is composed of a formal acceptance report (in Word) followed by support calls and enhancement requests by phone and email.

Formal methodologies are not used. Processes tend to be ill-defined. One respondent had responses suggesting a well-defined process and flow, but most were much less formal. The respondent with the well-defined process had not started development, so this was a planned, not an actual, process.

Completion criteria were very informal. No tools were used to verify achievement of completion criteria.

*In summary, formats for information interchange are either high-level and informal (Microsoft Office artifacts with no formal semantics) or semantically precise but very low-level (XML and code). High-level, semantically precise models are rarely used.*

*Tools are mostly non-domain-specific and low-level.*

*Reported processes are informal.*

### **6.3 Design Processes Supported by Tools Vendors**

The overall process is shown in the previous section, Design Processes Utilized by Technology Developers.

Tool vendors expect the same types of artifacts to be used in communication between processes. High-level, informal artifacts that were generated for use by humans (not tools) include documentation text in Word and HTML format. Low-level, precise formats imported from other tools or written out for use by tools include XML descriptors, Interface Definition Language (IDL) Common Object Request Broker Architecture (CORBA) interfaces, and code. One respondent reported reading XML Metadata Interchange (XMI) from modeling tools but no details were supplied.

*Tool developers report using the same interchange artifacts as technology developers. They may desire interchange through models, but they do not work at that level today.*

## 6.4 Key Issues

### 6.4.1 Technology Developer Key Issues

Three key issues were identified by technology respondents:

- Bad specifications  
*Does this mean product specifications or standards? The response was not clear.*
- Tools with inadequate capabilities
- Lack of interoperability between tools at different stages
  - Inability to import and export artifacts

*The main communication formats are text (requirements and change requests) and code or descriptors. There was very little in between. There was one reference to a model, highlighted by this quote:*

*“A larger problem is lack of a common language between the developers (those making interpretive decisions) and the end-users. Developers talk about bits and bytes, sample rates and so-on, end-users know that when they push a button something needs to happen. A systems engineer that is eloquent in both vernaculars is required to bridge this barrier. This expertise is rarely available on many jobs.”*

***It is difficult to see how interoperability can be achieved using informal communication artifacts. This may be the most significant observation from the RFI.***

The degree of pain associated with these issues was “Inconvenient.”

*This low level of pain is slightly surprising; perhaps these issues are considered to be unavoidable properties of system development.*

The respondents were asked to identify the solutions they’d most like to see. The only concrete response was, “Seamless tool integration from design to implementation.”

*This reflects the interoperability pain above, and is related to the same root cause: the absence or non-use of interchange formats with semantic meaning.*

### 6.4.2 Tool Developer Key Issues

Tool developers had some of the same concerns as their technology developer customers. The key issues they identified concerned standardization: “ambiguous, under-specified, and inconsistent standards” and “lack of consistency in the implementation of standardized SDR frameworks.” The level of pain, again, was “inconvenient.”

*Perhaps the level of pain is not higher because tool vendors concentrate on piece-wise integration against a small number of de facto standard tools.*

Reflecting these issues, tool developers would like to solve or have solved basic standardization issues. They would like to “document the common understanding of ambiguous aspects of standards and provide copious examples” and “[develop] more

rigorous testing and validation suites for these frameworks,... training materials and radio simulation tools”.

## **7 Waveform Development**

The Waveform Development section of the DPT-WG RFI explored the design processes associated with the creation of all or part of a waveform’s functionality, and the tools supporting this creation. For the purposes of the RFI, waveform development could occur on a specific radio platform, but the involvement by the waveform team in specifying and implementing this platform was outside the scope of the waveform development processes. The results of this section of the RFI follow.

### **7.1 Demographics**

The SDR Forum Design Process and Tools Working Group received twelve responses to the waveform development sections of the RFI: eleven from technology developers and one from a tool vendor. Of the eleven technology providers, eight considered themselves “Radio Set Providers,” with seven indicating they were involved in Tactical Military Communications and one in Commercial. The other three technology providers listed themselves as “Other Technology Developers (R&D),” with one in Military Satellite Communications, one in Commercial Satellite Communications, and one stating they were involved in “all of the above”.

Four of the respondents indicated that their companies had more than 5,000 employees, three respondents indicated that their companies had between 500 and 5,000 employees, and the rest indicated fewer than 500 employees. Team sizes varied, with three responses indicating fewer than five people on a team, three responses indicating between five and fifteen people on a team, two responses indicating between fifteen and fifty people on a team, and two indicating more than fifty people on a team. *There did not appear to be a strong correlation between company size and team size.*

Four companies indicated they had a dedicated tools specification group, but only one company indicated an in-house tools group. All the responses indicated that tool selection was done on a project-by-project basis, with one response indicating that tool selection was performed at the corporate level.

Only one tool provider responded to the RFI in this section, and it was not possible to draw conclusions from this single response. As such, the demographics associated with this response are not presented here.

### **7.2 Design Processes Utilized by Technology Developers**

An analysis of the RFI responses from waveform developers on their design processes yielded the state diagram shown in Figure 5. The process generally begins with a requirements specification that is usually supplied by the customer in the form of a Microsoft Word or Adobe PDF document. One or more MATLAB models or Waveform reference designs developed by the customer may accompany the specification. Once the requirements are received, the first step in the waveform development process is usually



Artifacts from the algorithm development stage are generally used to create a non-real time prototype of the waveform operating on a PC or workstation. The input and output for this “golden waveform” are generally supported through files that are generated and analyzed offline. Feedback from this stage may include modifications to the component model or the associated algorithms, and may also drive modifications in the requirements specification. Artifacts from this stage generally include the source code for the golden waveform, assembly descriptions for the waveform components, and a preliminary detailed design document using Microsoft Word, Visio, and Excel. A demonstration showing non-real time operation of the waveform under development may accompany these artifacts.

The RFI responses indicated that both the algorithm development and waveform prototyping stages could provide input to the definition of the actual radio platform. This input could include spectral masks, anticipated signal-to-noise ratios, and others. In return, the specification for the radio platform, including the configuration of the processing devices and the performance benchmarks of the data transports between devices, acts as a primary input in the next stage of waveform development: the creation of a deployment model. In this stage, the waveform components are partitioned across the various processing devices (FPGAs, DSPs, GPPs, etc.) based on both the platform specification and the supplemental specifications for the deployed radio, including power. Artifacts from this stage drive the requirements specifications for application code that runs on each device, and may require changes in the non-real time prototype model to accommodate operation on the target hardware. These artifacts may also provide feedback to the radio platform specification.

The final stages of the waveform development process are often the longest in duration: developing and testing the actual waveform code that is to operate on the radio platform. Core technologies, such as FPGA IP cores, can feed this stage, as well as provide artifacts such as MATLAB models that can be used in previous stages. The real-time debugging of the deployed code may yield feedback both to the radio platform specification or to the waveform partitioning model.

### **7.3 Design Processes Supported by Tools Vendors**

There were insufficient responses from tool vendors to provide meaningful results in this area. This is an area that requires further expansion going forward.

### **7.4 Key Issues and Future Directions**

*One of the key issues that came out in the analysis of the RFI responses was the dichotomy of views as to “what is a waveform.” People involved in the first few stages of waveform development had a very software/algorithm-centric view of the waveform, whereas people involved in the latter stages had a very hardware/embedded systems view of the waveform. The gap between these two views was striking, with both sides implying that the other side just does not understand what they have to do. This appeared to be at least partially due to the fact that the software developers use a different language to describe elements of their design than the embedded systems engineers use to describe*

*equivalent elements. This was illustrated in follow-on discussions related to components and modules, with software developers indicating that components are made up of modules, and hardware developers indicating that modules are made up of components.*

Issues that were relevant to tools in the RFI responses included:

- The inability of tools to support interaction within a multidisciplinary team
- The interoperability between tools at different stages, that is, the inability to import/export artifacts between tools
- The inability of tools to generate and tracking meaningful documentation
- The lack of support for real time debugging and real-time validation for waveforms operating across multiple heterogeneous processors

All of the issues were listed as “inconvenient but could be worked around,” although one response did have the following observation:

*“Many of the FPGA code generation vendors show a small scale demonstration. Most of the more complicated problems show up when doing a large-scale project so the benefits of these tools are not in touch with reality.”*

When asked the question “Without constraints (money, time, resources, etc.), what problem would you have someone solve to make your life easier?” a number of telling responses were received, including:

- “Model-based tools that could
  - allow expressive, and comms-specific modeling of communications designs;
  - predict (estimate) functional and execution performance for specific platform choices / implementation technologies;
  - allow IP (hand-coded modules and libraries) to be integrated in with the models;
  - generate the code that is repetitive (e.g. CORBA wrappers), while integrating with the critical hand-coded modules (IP); and
  - provide coherence between design documentation, simulations, and documentation.”
- “Better tools for C code to FPGA transition.”

## **8 SDR Platform Hardware Development**

The SDR Platform Hardware Development section of the DPT-WG RFI explored the design processes associated with the creation of all or part of the radio hardware (RF, mixed signal, digital) platform on which the waveform applications run, and the tools supporting this creation. The results of this section of the RFI follow.

## **8.1 Demographics**

The SDR Forum Design Process and Tools Working Group received twelve responses to the platform hardware development sections of the RFI: eight responses from software radio technology providers, and four responses from tool developers.

### **8.1.1 Technology Providers**

The breakdown of eight companies from the Radio Technology Providers space, according to their primary function, was as follows: six SDR technology–radio set providers (three tactical military communications (MILCOM), one commercial infrastructure, one radio frequency identification (RFID), and one choosing “all SDR applications”), one end user, and one in research (RADAR field).

### **8.1.2 Tool Providers**

The breakdown of the four respondents from Tool Developers space, according to the companies’ primary function was as follows: one commercial cellular, one SDR technology–radio set provider (commercial wireless communications and services), one choosing “all applications,” and one choosing “other” (FPGA vendor: tactical MILCOM and commercial wireless).

## **8.2 Design Processes Utilized by Technology Developers**

Analysis of the RFI responses from the technology developers produced the state diagram shown in Figure 6, which describes their design processes. The input to the process is a waveform specification supplied by the customer in one of various forms (MS Word document, MATLAB file, SCA wavform binaries, SCA core framework binaries). The first step in the process is to develop the algorithm and simulate the waveform. *The next step in the process is defining the architecture to be used to implement the algorithm. A key part of this step is identifying the IP blocks (either hardware or software) to be used. This may involve a development effort if they are to be developed in-house, or identifying third-party IP providers. The form of the IP will have platform partitioning implications downstream.* The waveform is then simulated non-real-time to ensure functionality and cycle-accuracy. Once the waveform is functionally correct and cycle-accurate, it is partitioned across the different processing engines (e.g., GPP, DSP, FPGA). The final step in the process is to verify the performance and functionality via real-time system verification on the hardware platform. The entire process is highly iterative in nature with design reviews after each step that provide a mechanism for feedback to every step in the process (e.g., algorithm development, architecture development, waveform partitioning, system verification). The process is usually complete upon passing a final design review, including a design acceptance test involving a technology demonstration and final assessment of system performance (e.g., cost, area, speed, power), with feedback to the requirements specification indicating trade-offs and proposed changes (e.g., functionality issues or feature suggestions) made using Microsoft Word, PowerPoint, and Visio. These artifacts are generally stored in some type of project folder that is tied to a requirements management system for requirements traceability.

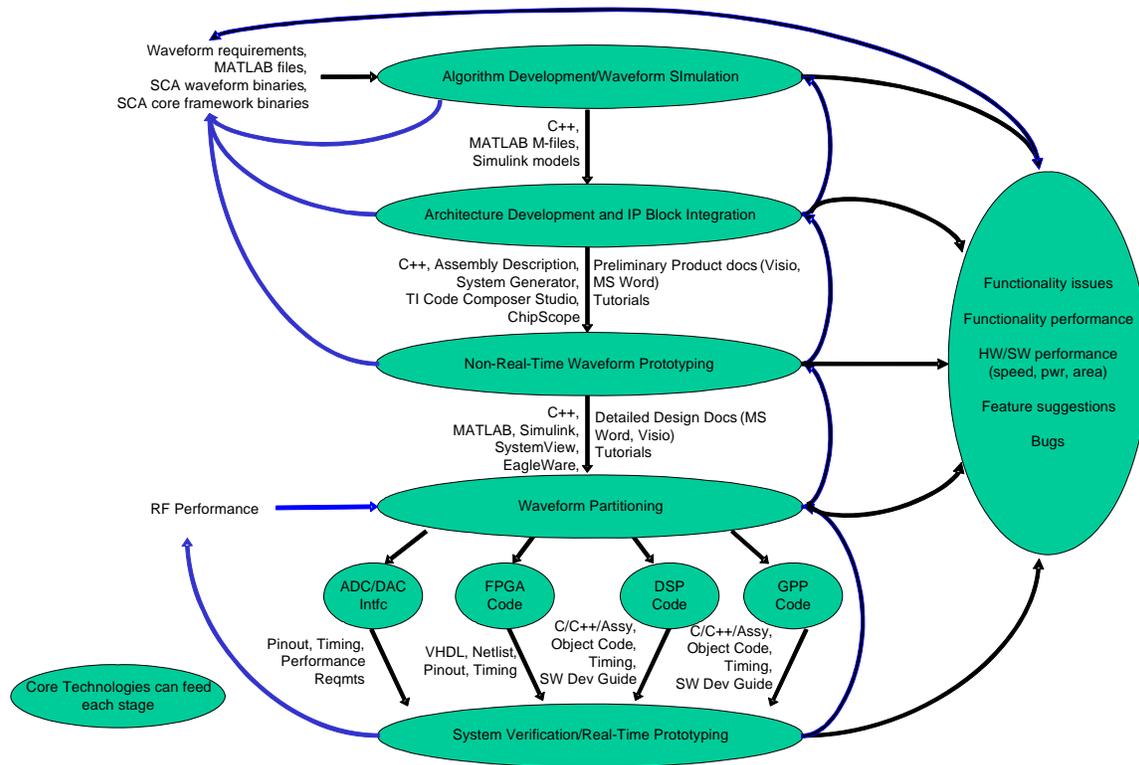


Figure 6: State diagram detailing the platform hardware development process

### 8.3 Design Processes Supported by Tools Vendors

Although there were an insufficient number of responses from tool vendors to draw any major conclusions in this area, a couple of interesting responses were obtained:

- One respondent mentioned “access to classified waveforms” as a major difficulty in building tools to support this development.
- Another mentioned “managing the balance between programmability and reconfigurability.”

This is an area that requires further expansion going forward.

### 8.4 Key Issues

Issues that were relevant to Tool Developers in the RFI responses included:

- Common difficulty is access to waveforms (classified and unclassified).
- An apparent desire from the tool vendors to become more intimately involved – whether that is the development of IP, or the support for Integration and Test.
- No other commonality in responses. Other difficulties were “Tool Flow and End User Bias.” A GPP vendor described “parallel processing” as a difficulty.

Issues that were relevant to Radio Set Technology Providers in the RFI responses included:

- Processing element of choice are: DSP (6), FPGA (5), GPP (4), ASIC (1).
- Common Design Methodology is Iterative Approach: Build/Test/Modify. This is also a source of difficulty since too many iterations are required.
- MATLAB/Simulink was the most identified tool used.
- The gap between hardware and middleware—hardware abstraction layer (HAL), SCA, and CORBA—must be addressed.

## 9 SDR Platform Infrastructure Development

### 9.1 Demographics

The SDR Forum Design Process and Tools Working Group received four responses to the platform infrastructure development sections of the RFI: two responses from software radio technology providers, two responses from tool developers, and one responded to both sections (although listed as a radio technology provider).

#### 9.1.1 Technology Providers

The breakdown of the three companies from the Radio Set Technology Provider space, according to their primary function, were all tactical MILCOM.

#### 9.1.2 Tool Providers

The breakdown of the two respondents from Tool Developers space, according to the companies' primary function was as follows: one tactical MILCOM and one commercial.

### 9.2 Design Processes Utilized by Technology Developers

There were insufficient responses from technology providers to provide meaningful results in this area. This area of the RFI requires further expansion going forward.

### 9.3 Design Processes Supported by Tools Vendors

There were insufficient responses from tool vendors to provide meaningful results in this area. This area of the RFI requires further expansion going forward.

### 9.4 Key Issues

*One of the key issues that came out in the analysis of the RFI responses was the lack of integration of the various tools in the SDR chain and the problems associated with version compatibility. The consensus was that an area of emphasis should be on the development of tool interchange standards and the development of end-to-end processes and tools to support the development of SDR applications, components, and platforms. The tool developers equally struggled with the same issue and mentioned a lack of consistency in the implementation of standardized SDR frameworks as a major impediment to the development of SDR tools.*

Specific issues that were relevant to technology providers in the RFI responses included:

- All respondents had issues with the processing element of choice.
- All respondents said that a lack of common design methodology was an issue and provided three unique responses as to methodologies they use:
  - Software Engineering Institute (SEI) Level 3/4 methodology along with commercial off-the-shelf (COTS) integrated development environment (IDE) toolsets integrated together.
  - Component-based programming (CBP), UML-based design.
  - Object-oriented analysis, object-oriented design patterns.
- Information interchange between tools has not been standardized sufficiently.
- The standardization of formalized metamodels/UML profiles of these different aspects leading to the capture of domain expertise in the areas related to the radio domain, and the integration of these metamodels into one consistent view.
- All respondents said that common difficulties were version compatibility and tool integration.

Specific issues that were relevant to tool vendors in the RFI responses included:

- All respondents had issues with the processing element for which the tool provides a solution.
- No other commonality in responses – the following represent individual responses:
  - Requirements
  - The reconciliation of orthogonal aspects of the radio platform

## 10 RFI Summary

Respondents to the survey identified inadequate tool interoperability as a key issue. The most widely used tool suite in SDR technology development today is Microsoft Office. These two recurring themes are strongly related.

Respondents repeatedly identified a set of closely related issues:

- Reported SDR development processes today are mostly informal or ill-defined.
- Standards for exchanging information between different steps in processes are not mature.
- Tools do not use standard interchange formats and are not interoperable.

Most development tools and the artifacts they produce are not domain specific. They are either very high level and informal (such as documents created by Microsoft Word) or very low level (such as C compilers and code). The information exchange artifacts are usually the informal ones: Word documents and the like. This is a fundamental barrier to transmitting design artifacts between different stages of development. As stated above, “It is hard to see how interoperability can be achieved using informal communication artifacts.”

The communication problem is especially acute between interdisciplinary teams. Software and hardware people speak different languages and have different perspectives. This is not just a notation or tool problem; it is an outlook problem.

The standards and communication problems are all related. A slightly different key issue is poor support for real-time debugging and validation on systems supporting multiple parallel processing devices.

These issues are mostly “worked around” with varying degrees of success. Survey respondents repeatedly identified the same set of issues, but they rated them as irritants, not show-stoppers. As software defined radios and waveforms are being built and delivered today, this should not be surprising.

## 11 Next Steps

Even though the results of the RFI provide significant insight into the processes and tools associated with the creation of SDR technology, with some of the associated issues, additional data is necessary to develop a more thorough model. Methods for obtaining this data were discussed by the SDR Forum Design Process and Tools Working Group, including issuing another, more focused, RFI.

The method for expanding this work that was finally agreed upon was to develop a Design Process and Tools Wiki Page that will be attached to the SDR Forum web site. The process for maintaining the page will be aligned with the SDRF processes as follows:

- Baseline content of Wiki pages will be taken from this RFI final report once it is approved by the Forum.
- Moderated “change proposals” from the SDR community will be added as “working” Wiki pages.
  - The Wiki page will be assigned a W document number.
  - A tree structure will be established.
  - A message board will be established for each new page.
- Information from tools vendors on how their tools will support a specific design process will be included as “input” Wiki pages
  - They will be assigned an I document number

As best practices emerge, they will be put forth by the Design Process and Tools Working Group for “approval.” This concept was presented to the SDR Forum Technical Committee, and the concept was approved, with a goal to have the prototype Wiki Page “go live” at the SDR ’06 Technical Conference.

## Appendix 1: Definitions

The following presents the definitions of terms that were used in the RFI:

**Air Interface** - The subset of waveform functions necessary to establish communication between two radio terminals. This is the waveform equivalent of the physical layer and data link layers following the Open Systems Interconnection (OSI) model.

**Application** - One or more components of a system that performs a specific task. The software necessary to perform this task is referred to as a software application. In the context of SDR, the term "application" includes software, firmware, etc. that define the air interface, modulation and communication protocols, or any part of the waveform, and manages or controls the radio in a network environment.

**Component** - Represents a modular part of a platform that encapsulates its contents and whose manifestation is replaceable within its environment. It exposes one or more ports, and its internals are hidden other than as provided by its interfaces. The ports define the component specification in terms of provided and required interfaces. Component applications are constructed by connecting ports.

**Model** - A representation that can be communicated and understood in specifying the functionalities, requirements, and structures of components, or the modes and configurations of an SDR system in an unambiguous and verifiable manner. A model may be used to design, implement, test, and validate all or part of the SDR system.

**Port** - An interaction point between a component and its environment. A port may provide interfaces to its environment, require interfaces of its environment, or do both. A port is directed: two ports may require or provide the same interface, but the component recognizes them as distinct interaction points. Component implementations see ports as their external interfaces and are encapsulated from the external port connections. Interfaces are defined by the protocol that implements the port. They are typically sets of operations or messages (either synchronous or asynchronous) or data streams.

**Radio Platform** - A collection of hardware, software, and firmware components that have been integrated to support one or more target applications.

**Radio System** - A collection of integrated radio platforms running one or more applications that solve the problem of the end user or users.

**RF Signal** - The actual transmitted "RF waveform" in the air interface; an analog representation that has characteristics defined by the RF waveform.

**RF Waveform** - The representation of a signal as a plot of amplitude versus time. This is what was traditionally referred to as a waveform.

**Validation** - Confirmation, through the provision of objective evidence, that the requirements for a specific intended use or application have been fulfilled.

**Verification** - Confirmation, through the provision of objective evidence (e.g., unit testing), that specified requirements have been fulfilled.

Waveform - Term used to describe the entire set of radio functions that occur from the user input to the RF output and vice versa. Examples are single-channel ground-to-air radio systems (SINCGARS), Association of Public Safety Communications Officials (APCO)-25, and the Global System for Mobile communications (GSM).

## **Appendix 2: Raw RFI Data**

The raw RFI data is captured in a Microsoft Excel Spreadsheet entitled “DPT RFI Response.csv”. This spreadsheet is stored in the Design Process and Tools SharePoint directory and can be made available upon request.