ECIX QuickData Architecture

(Internet B2B e-Commerce and XML-based Component Information Exchange)

Goal of the ECIX Project

The Electronic Component Information Exchange (ECIX) project was originally dedicated to designing standards for B2B technical information exchange of component information for traditional electronic components (i.e. the EC domain). The ECIX QuickData architecture can support other domains such as virtual component (i.e. the SoC domain) and other domains easily using domain-specific dictionaries. QuickData architecture, technology, and standards are extensible, unambiguous, well documented and are maintained under the direction of Si2.

QuickData Specifications

The QuickData Specifications that implement this architecture enable electronic component and SoC customers and other component information end users to make standards-based queries to compliant suppliers over the Internet, and receive real time responses in a standard format. Business-to-business (B2B) technology must be a major element in every company's Internet strategy, and QuickData meets key requirements of participants in both the traditional electronic component and SoC component domains. Readers are encouraged to review the QuickData Information Set for direct links to all specifications and DTDs. Also see the QuickData 2 Specifications for an overview of the documentation set for QuickData 2.

The QuickData architecture (Figure 2 below) was designed to enable real-time B2B communications (using the Internet and based on industry standards such as XML) between all of the participants involved in the design and manufacture of electronic systems. This includes original equipment manufacturers, component manufacturers, distributors, value-add information providers, and EDA companies. Each of these companies well understand the need to minimize time-to-market, and are thus motivated to reduce cycle times for each of the key phases of product development and manufacture.

The Engineer's Problem

In Figure 1 below, the scope of the ECIX project is described at a high level. Product

development at any packaging level goes through a process similar to the conceptual steps below. These steps are not necessarily serial, but each step will eventually be completed:

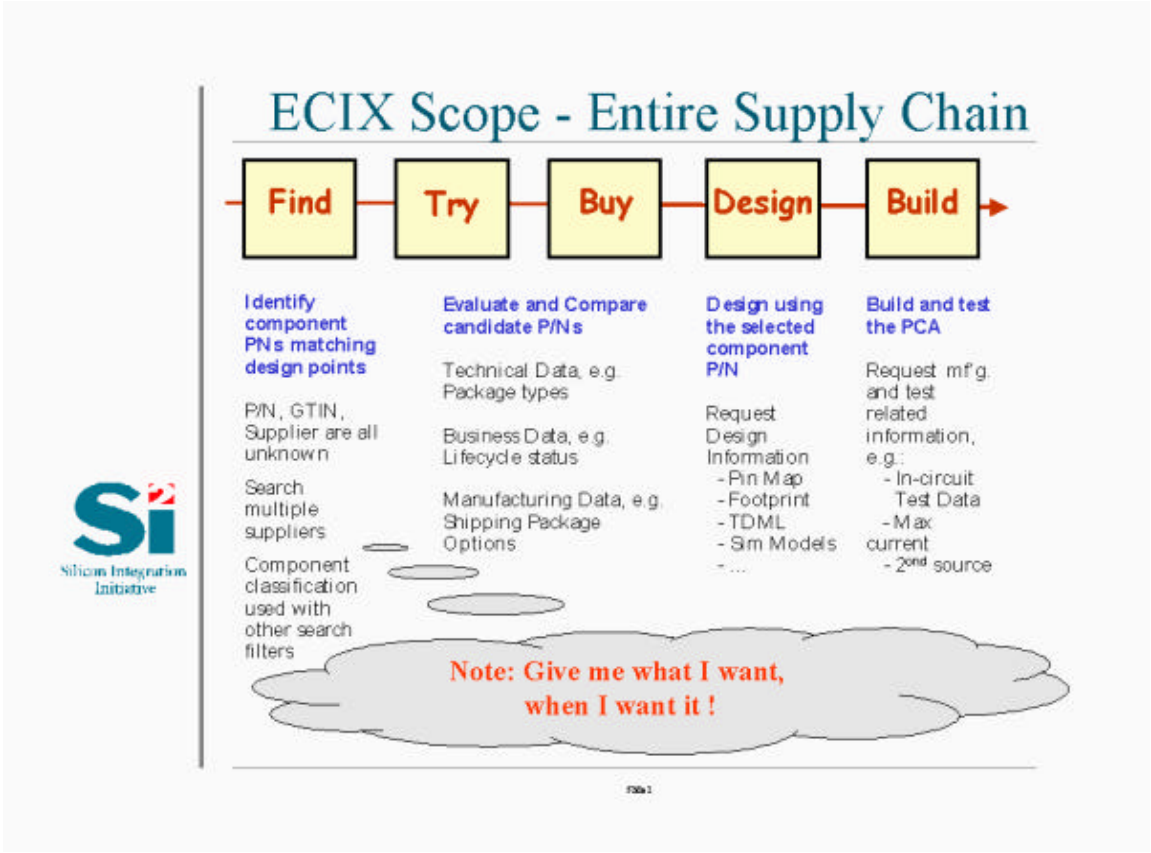

Figure 1 - The Scope of the ECIX QuickData Architecture -- the complete EC Supply Chain

Find -- Before real product design can proceed in any detail, the key building blocks in the design must be identified. To find candidate components, a process of searching for components is involved, looking for those components that match key parametrics related to the design problem. Here, a customer design or procurement engineer is searching for components (traditional or virtual components). In this phase, the customer may use a graphical user interface (or other software) to develop a query for components that meet certain parametric values (properties or attributes). For example, an engineer might be looking for a memory device, in DRAM technology, with 32-bit addressing, and with access time less than some value. In this phase, the designer wants to search the industry for the best component available that matches the design needs. The designer may or may not know which manufacturers (component suppliers) provide the desired component. In the "find" phase, the designer identifies candidate components, and then selects one (or more) for further evaluation.

Try -- Once a candidate set of components from known suppliers are identified that match the key parametrics specified by the designer, the real task of technical evaluation of the component begins. To accomplish this technical evaluation, additional information may be required from the component manufacturer, such as the component datasheet, high-level simulation models suitable for system simulations, etc. In the "try" phase (or comparison phase), this information is requested, delivered, and evaluated.

Buy -- In the "buy" phase, the candidate components are evaluated from the business perspective; i.e., information such as lifecycle information, price, availability, and other important component characteristics are evaluated. This phase is also considered part of the "comparison phase".

Design -- In the "design" phase, the decision to use a specific component from specific supplier(s) has been made, and real product design using the selected component can begin. In order for the design process to be enabled, typical EDA design libraries must be available. Simulation models may be required for complex devices. EDA symbols for the customer's design methodology and toolset must be made available. Footprint information is required to support layout, etc.

Build -- Finally, the product design is complete, and the manufacturing process begins. Many types of data may be requested during this phase that relates to the manufacturing/test process (e.g. in-circuit test data, pick and place information, etc.). Many times this phase is out-sourced to a manufacturing company different from the product design company. It's not unusual for a component on product being manufactured to go "end-of-life"; when this happens, it generates a need to start the "find" process all over at the beginning, to find an equivalent component to replace that component.

The ECIX QuickData Architecture

In Figure 2, the ECIX QuickData Architecture is illustrated. This architecture was designed to enable the B2B transactions necessary to support the above customer-supplier relationship, and to support all of the phases of design; FIND, TRY, BUY, DESIGN, and BUILD per Figure 1. First, the various environments at the customer and supplier companies are discussed. Then, the architectural approach implemented in QuickData is described.

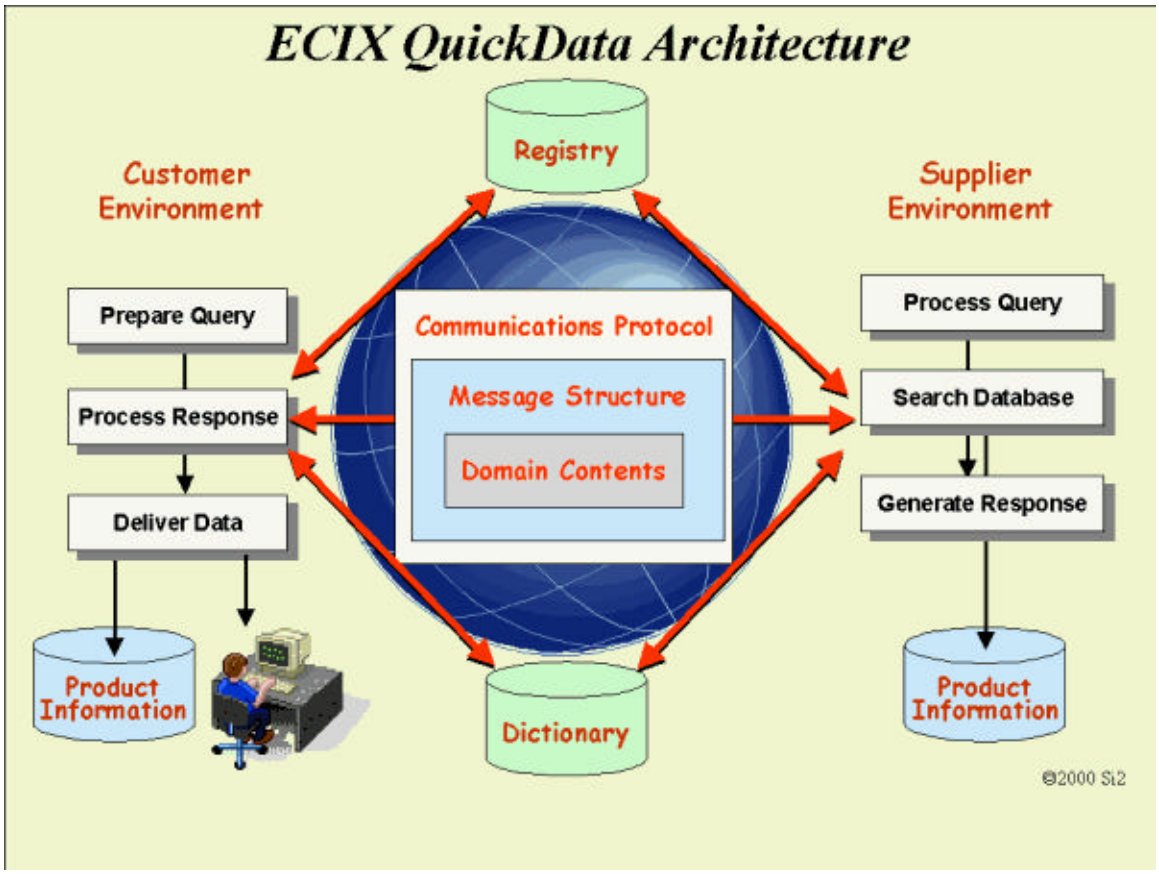

Figure 2 - The ECIX QuickData Architecture

Customer Environment

Many large customer companies have large component or product information databases (CIDBs) of "qualified components" which are preferred by the company to be used by their design engineers (presumably because of the company's relationship with those suppliers -- e.g., components have been tested and qualified, special prices, availability, etc.). In such companies, the methodology is for the engineer to first search in the companies CIDB when looking for components for new designs. Only when such a search proves insufficient are the engineers allowed to "look outside" the company for other candidate components. Even then, candidate parts must then be "qualified" and marked available for design use in the CIDB.

Other companies, particularly smaller companies, may NOT have a qualified database of preferred parts. These companies by definition must look outside their own company to find candidate parts.

Without a standard way of searching, all of these companies are forced to access the component information (web sites, printed materials, etc.) in a highly manual fashion, with the format and content of each supplier company being different! This is a very difficult usability problem for

the engineers in these companies, who see no value in the "unique appearance" of each supplier's web interface! The QuickData architecture addresses this problem head-on.

Customer Use of the Registry

Before a query can be sent, it must be determined where to send the query. Using the Registry (see below), the customer company can control the list of suppliers that the engineering or procurement engineers or other users can get access to. For example, the company may have more than one registry. One might be a registry of "qualified suppliers". Another might be a registry for a user to use if the qualified supplier registry fails to locate a suitable component. In any case, the customer can send a query once one or more suppliers have been identified. Then, the query is sent simultaneously to all of the selected suppliers (one or more). As used here, "suppliers" refers to suppliers of information about components (e.g., the manufacturer, distributors, value-add information providers, etc.).

Customer Use of the Dictionary

The dictionary for a given domain defines the properties (or characteristics) used in that domain. For example, in the EC domain, "pin.count" might be defined as the "number of I/Os for the component", which is an integer, one to four digits, etc. In the VC domain, VC.ID might be defined as "the unique identifier assigned to the VC by the provider". The goal of the dictionary is to unambiguously define the domain properties so that the customer (and the supplier) knows precisely what information is being requested in a query and how to interpret that information.

Note: While customers can query for characteristics that are not properties in a domain dictionary and still be compliant, the intent is for all properties to be defined in a dictionary. Customer queries which contain characteristic values for properties not defined in the domain dictionary are typically used only for piloting queries for new information not yet defined in a dictionary, or to exchange information between consenting companies that is not (yet) part of the standard dictionary for that domain.

Prepare Query

Enter ECIX QuickData; the typical scenario is a customer using a graphical user interface (GUI) of a company proprietary query engine, or to any other query engine. Using the GUI, the customer creates a query with the desired characteristics for the desired component. From the example described in "Find" above, a customer might be looking for a memory device, in DRAM technology, with 32-bit addressing, and with access time less than some value. Queries would typically be built for any property, property set, or product information object (PIO) in the domain-specific dictionary (in this case, for the EC domain). See below for additional information on PIOs. In fact, the architecture supports queries for any item, whether in the dictionary or not! This allows B2B transactions between consenting companies to occur, in a compliant way. Strategically, all items in a query or response should be in the dictionary, but that is not absolutely required to do business the QuickData way. This approach allows new properties, etc. to be requested and delivered before they are officially part of the standard

dictionary. Companies can do business and propose extensions without waiting for the official dictionary updates to occur.

Process Response

As the responses to a query are received back from each of the selected suppliers, those responses are processed by customer software. The possible responses range from a list of matching components (from each supplier), to a time-out because a supplier server was down, etc.

The customer's response handler parses the XML responses, and acts on those responses in any manner desired by the customer. For example, the responses could be sent to an engineer's screen (e.g., query GUI software), or delivered to a software program (such as an EDA application), or the data could be delivered automatically into the customer's product information database.

Deliver Data

For example, the information received back from the suppliers could be collected and delivered to a browser screen. This information is typically presented in a form that enables the customer to easily compare important properties of each of the candidate components. Upon inspection, the customer may decide to request additional information such as a datasheet for the selected component from the selected vendor (in QuickData architecture, a datasheet is one of several files a vendor may have available for each component; these files are called "product information objects").

If an additional product information object (PIO) is requested, then a specific query is sent to a specific supplier for the desired PIOs. The response handler, upon receipt of the PIO, delivers that information to the appropriate handler for that PIO (i.e., via MIME type). For example, if a datasheet was requested for a component (datasheet.pdf), then the MIME-type could cause a launch of Adobe Acrobat (or any other application) to properly process the datasheet when it is delivered to the customer.

The response handler, in addition to presenting response information on a user's screen, or launching an application (such as an EDA application, etc.) on a returned PIO as described above, might also process the response and update some local company "product information" (or CIDB).

Product Information (Customer)

Some companies may use this approach to deliver information to "auto-load" their CIDB with selected characteristic values, property sets, and PIOs for components they want in their qualified component database. There is no requirement and no standard for specific tools or what customers should do with the information received from the suppliers based on queries. The only relevant standards are the format for the query and the response, and the protocol to be used during the interchange between the companies.

Supplier Environment

Supplier(s) typically have a "product information" database that contains all the information (including datasheet PDFs) for the component products offered by that manufacturer. The key to the QuickData architecture for suppliers is to make the information available to the customers in an unattended and automatic fashion, in real-time. In QuickData, the supplier has the architectural elements described below.

Supplier Use of the Registry

Before a response to any query is prepared and sent, the supplier might want to evaluate specifically who sent the query. Using the Registry (see below), the supplier can locate additional information regarding who is making the query from the list of customers registered. If the customer making the query is registered, then the supplier can use that information to decide specifically what information to return to that customer based on their business relationship. If the requesting customer is a very important customer, then the supplier may decide to return special information in response to queries. If the requesting customer is not so important, or is not even in the registry, the supplier may choose to return less information in response to queries, or to respond with a special message code, or perhaps not to respond at all.

For example, the supplier may make use of more than one registry. A registry might be the open industry Si2 registry. Another might be a registry of "important customers". In any case, a response to a query might be tailored to "who is asking".

Supplier Use of the Dictionary

The dictionary for a given domain defines the properties or characteristics used in that domain. For example, "pin.count" might be defined as the "number of I/Os for the component", which is an integer, one to four digits, etc. In the VC domain, VC.ID might be defined as "the unique identifier assigned to the VC by the provider". The goal of the dictionary is to unambiguously define the domain characteristics so that the supplier (and the customer) knows precisely what information is being returned in a response and how to interpret that information.

Note: While suppliers can return characteristics that are not properties in a domain dictionary and still be compliant, the intent is for all properties to be defined in a dictionary. Supplier responses which contain characteristic values for properties not defined in the domain dictionary are typically used only for piloting responses containing new information not yet defined in a dictionary, or to exchange information between consenting companies that is not (yet) part of the standard dictionary for that domain.

Process Query

On the supplier side, QuickData queries are handled much like a mail handler. When a query is received, the supplier's QuickData query handler software is invoked. The query is "parsed" to determine who the customer is, and what the customer is asking for. It is important to note that

the supplier can determine specifically "who is asking", and thus provide appropriate responses to that customer based on their business relationship. For example, two identical queries received by a supplier from two customers might each request a simulation model for a component. One of those queries is from a very large customer with which the supplier has a special relationship, and thus allows the simulation model to be delivered to that customer. The same query received from a very small customer to that supplier might receive a response of "not available to you". It is up to the supplier to determine which response to provide to the customer, based upon the information available and the business relationship with that customer. Both of the responses described below can be delivered in a compliant manner.

Search Database

Assuming the supplier wants to respond to the customer, there are two general cases to handle (possibly both at once).

The first kind of query is a customer request to "search for components that match the following characteristic values". Again, we can use the example of the customer looking for a memory device, in DRAM technology, with 32-bit addressing, and with access time less than some value. Each of these properties can be easily described in a QuickData 2 XML query. The supplier parses that query, figures out what the customer is asking for, converts the QuickData 2 query into the appropriate form so that the specific "product information" databases can be searched.

The second kind of query is when the customer has already selected a vendor and a specific component, and the customer now wants access to additional product information objects (PIOs). In such queries, the customer provides the component part number identifier (supplier part number or GTIN, and/or other characteristic values) and the desired PIOs. The supplier parser then decodes the query XML, figures out which PIOs the customer requested, performs a search in the supplier product information base, locates the desired characteristics or PIOs, and if found, delivers them back to the customer in real-time. If not found, appropriate characteristic or PIO null value response or other return codes are set for each, and returned to the requestor. Additional information on product information objects is located below in the section on Electronic Component domain PIOs.

Product Information (Supplier)

The supplier typically stores all of his product information in a database or other collection of data files. This information base may be queried for a variety of information objects such as searching for parts, searching for files, etc. There are no specific requirements on the supplier's database technology; the only requirement is that parametric searches (i.e. searches for components with certain values of properties (e.g. access time less than some value for a memory device) may be performed. Also, searches for PIOs (product information objects) such as datasheets (or simulation models, etc.) must be supported. Typically, this kind of capability is provided in query-based database products.

The supplier may have an Oracle or DB2 or other database, or other solution to storing

component information. In any case, the customer's query is converted to the appropriate database query, which is sent to the supplier's specific database in an effort to find matching parts. The result of that search may be one or more components that match the query.

Generate Response

The response data for each of those matching components is then converted back into a standard QuickData 2 XML response, and returned to the customer for processing.

Registry

The Si2 Registry is a computer-sensible listing of companies and products that are QuickData compliant. Companies may register as customers, or suppliers, or both. Software companies may also register software such as query tools in the Si2 Registry. Si2 is collaborating with RosettaNet to help define a more general RosettaNet registry. When additional information is available regarding the RosettaNet registry interface, it will be posted on the RosettaNet site.

The Registry is a key element of the QuickData architecture that identifies "who is playing in the game". When customers are searching for components or looking for characteristics of specific components or specific product information objects for a component, they need to know specifically where to send such queries (i.e., the URL of the supplier sites). This specific information that identifies the supplier(s) and the URL of the QuickData server is contained in the entries for each registered supplier. Similarly, customers get registered as well. See the Si2 Registry Specification for details.

Companies may now register in the official Si2 Registry. See the QuickData Registry Process on the web for details! Current companies in the registry can be viewed at the Registry listing! For additional information on the Si2 Registry, see the April 10 announcement!

Dictionary

In the QuickData architecture, the goal is that each characteristic or PIO that a customer or supplier might want to reference in a query or a response is defined in a computer-sensible manner, and in a prescribed format. This computer-sensible dictionary is designed to enable this terminology and additionally, for a component classification scheme to be unambiguously described. For each property or PIO defined in the dictionary, a name, a definition, the value format and units are unambiguously defined.

The QuickData 2 Specifications now support the new RosettaNet (and Si2) Electronic Component Technical Dictionary (ECTD) Specification format for computer-sensible dictionaries. The QuickData protocol specifications state that each domain-specific message be supported by a dictionary, such as *any* ECTD instance. This architecture is designed to enable requirements for new information flow between companies to be implemented without long development cycles. When new data objects or PIOs (e.g. existing legacy data files) or characteristic values are needed to be exchanged between companies, the information need only

be entered into the appropriate domain dictionary, and the technology works immediately! The Electronic Component Technical Dictionary (ECTD) Maintenance Specification describes the process by which ECTD changes get managed at Si2. See the Electronic Components Domain Specifications below for important additional information regarding dictionary instances in the ECTD format.

The Si2 Electronic Component Technical Dictionary (ECTD) Instance (right click to download) is a sample of the computer-sensible QuickData properties for the EC domain.

Note: the ECTD instance above currently contains only Si2 QuickData 2 required content. RosettaNet plans to officially release (very soon) the RosettaNet ECTD instance that contains the QuickData required elements. QuickData 2 is designed to use the RosettaNet ECTD instance (which is the result of combining the dictionary work of IEC, EIA, JEDEC, ECALS, Si2, and RosettaNet) as soon as it is available. See RosettaNet for information on availability of the RosettaNet ECTD instance.

Protocol

When the customer has generated a QuickData query, it is sent to the selected suppliers (selected from the Registry). There are many possible methods that could be used to "wrap" that query (or message). Both RosettaNet (for PIP™ 2A9) and ECIX QuickData use standard HTTP protocol, and CGI-Post method for transmitting the message between the servers of the companies involved in the transaction.

The current protocols supported by the QuickData architecture are specified below. Follow the links for the detailed specifications of each of the protocols described below.

RosettaNet Protocol (RNIF 1.1)

The RosettaNet Protocol defines the specific technologies used for the communications protocol between companies compliant to RosettaNet Interface Framework, i.e., HTTP, CGI-POST, SSL, etc. The RosettaNet Protocol is implemented via HTTP, using SSL and CGI-Post method. See the RosettaNet Implementation Framework Specifications (RNIF 1.1) at the RosettaNet web site for more information on this protocol.

The Si2 ECIX team is excited to announce that RosettaNet has an agreement with Si2 for the development of Partner Interface Process™ 2A9, Query Electronic Component Technical Information. PIP2A9™ ; will utilize all major QuickData components and will be developed in a phased approach. Specifics of this phased transition plan are outlined in schedules for planned enhancements! There are major activities planned leading to exploitation of the new component classification taxonomy contained in the RosettaNet Electronic Component Technical Dictionary (ECTD) later this year. Additional information on the plan details for PIP2A9™ Version 2 will be placed on this web site when available.

The ECIX QuickData 2 release described below is designed to be RosettaNet compliant and adds

important new functionality, while minimizing migration impact for implementers already invested in prior releases of QuickData. A zip file of the primary QuickData 2 documentation set is available. All appropriate documentation will be transitioned to RosettaNet during the development of PIP2A9™ .

QuickData Protocol

The QuickData Protocol Specification defines the specific technologies used for the communications protocol between companies compliant to QuickData 2 (i.e., HTTP, CGI-POST, SSL, etc.). This specification also defines the specific message *format* (i.e., DTD) to be used for all non-RosettaNet QuickData application domain-specific messages (such as those below, under Electronic Component Domain and QuickVC Specifications).

Message Structure

The Si2 QuickData 2 Message Specification describes the message structure and constraints on messages. Generally, the QuickData architecture supports a broad range of application domains by specifying that all messages have a common structure. For example, all properties have a "name" and a "value". The message structure states that for the Electronic Component (EC) domain, "pin.count" is a property name. The definition of pin.count is specified in the EC Dictionary (i.e., the ECTD from RosettaNet), and it has a value format of "integer, one to four digits". This message structure was chosen to make maintenance simpler (no DTD changes required if new properties are required), and to enable computer-sensibility via the dictionary without having to encode all data in the message itself. For example, the information in this example that "pin.count" is an integer does not have to be coded in the message itself, since it is described in the dictionary.

See the Si2 QuickData Message Specification for details.

Domains (Specific Domain Messages)

There are currently several QuickData domain specifications, as described below. Each of these specifications plays a different role in the QuickData paradigm. The specific message content for the Electronic Component Domain and QuickVC Specifications for the SoC domain follow the rules of the QuickData Protocol Specification and the QuickData Message Specification. Also see the Product Information Object (PIO) list for a list of object specifications. Support for other domains is also possible. There is potential for additional Si2 domain support based on QuickData technology, and such support could potentially be transitioned into RosettaNet based on agreed-to plans.

New specifications for new domains can easily be created by defining a new dictionary of properties for the domain.

The dictionary for the domain contains, for each property, or data object, or component class:

- The name
- The symbol
- The definition
- The format (value type, units, conditions)
- Information regarding whether the item is required or optional in QuickData queries and responses.

QuickData Technology for the EC Domain (Electronic Components)

The Electronic Components Domain Specifications address the traditional electronic components for PCB domain. These specifications define the specific QuickData message *content* (as opposed to format) used in this domain. The content relevant to the traditional component domain is precisely and unambiguously defined in a computer-sensible EC dictionary.

The ECIX QuickData 2 release described below is designed to be RosettaNet compliant and adds important new functionality, while minimizing migration impact for implementers already invested in prior releases of QuickData. A zip of the primary QuickData 2 documentation set is available. All appropriate documentation will be transitioned to RosettaNet during the development of PIP2A9™.

ECIX QuickData 2 Overview: this release will:

- Become the technology for RosettaNet PIP2A9™ Version 1 ("Query Electronic Component Technical Information") and be RNIF v1.1 service-content compatible (i.e., compatible with the actual PIP2A9™ electronic component domain technical information content, and the RosettaNet protocol wrapper content).
- Enable secure transactions via support for SSL and certificate authentication.
- Provide a high degree of compatibility with software and instances based on prior versions of QuickData. Most software can process the new DTD as before, simply ignoring the new tags or attributes. Existing software would have to be modified to take advantage of the new features.
- Address ambiguities and issues related to QuickData 1 specifications (QuickData 1.01 and QED 1.1).

Enable extensions planned for PIP2A9™ Version 2 planned for later this year. This major release will include support for component classification (taxonomy) and other important features.

The Si2 QuickData Application: Electronic Components Domain Specification describes in detail the messaging for the EC domain (Note: this specification will be superceded by PIP2A9™ when officially available). The Si2 Electronic Component Technical Dictionary (ECTD) Instance (right click to download) defines the computer-sensible QuickData terms for the EC domain.

Using PIP2A9™ and the QuickData EC Domain specifications, customers can query for electronic components with specified parametric values.

For example, with this specification, an engineer could develop a query for a 64M DRAM memory in a SIPP package, with an access time less than some value, and have that one query sent to multiple suppliers at the same time, with all matching responses obtained in real time for designer review.

Product Information Objects and Property Sets for the Electronic Component Domain

The EC domain includes support for several types of objects that can be requested via queries. Customers can now optionally query for any properties, property sets, or product information objects (PIOs) defined in the EC Technical Dictionary.

Product Information Objects:

- EDA Symbol (PinMap)
- EDA Footprint (ECPackage)
- Table of Contents (TOC)
- Interactive Timing Diagrams (TDML)
- PCIS (Pinnacles Component Information Standard)
- Si2 Registry information objects (DTD, Registry) (see the EC Dictionary instance for details)

Property Sets:

- Lifecycle Information (see the EC Dictionary instance for details)
- Moisture Sensitivity (see the EC Dictionary instance for details)
- Electrostatic Discharge (see the EC Dictionary instance for details)
- Risk Assessment (which includes Lifecycle, Moisture Sensitivity, and Electrostatic Discharge)

QuickData Technology for the SoC Domain (QuickVC)

The QuickVC Specification is for the SoC components (or virtual components) domain. This specification is still in DRAFT status. QuickVC defines the specific QuickData message *content* (as opposed to format) used in the SoC component domain (e.g., virtual components, or IP, for system-on-chip design). The content relevant to the SoC component domain is precisely and

unambiguously defined in a computer-sensible dictionary.

- QuickVC Specification (HTML) (Zip)
- QuickVC Samples (Zip)
- QuickVC Dictionary (PDF) (Zip of XML)

Note: The QuickVC specifications utilize Si2 QuickData technology.

Using the QuickData QuickVC Domain specifications, customers can query for SoC components with specified parametric values.

For example, with QuickVC, an engineer could develop a query for a VC in the class of "Viterbi", a market segment of "communications or consumer", a hardness of "soft", a maximum gate count of "30K", and a compliant standard of ETS 300 401". That query could be sent to multiple suppliers at the same time, with all matching responses obtained in real time for designer review.

At DATE 2000 (Paris) in March, the Nokia Research Center, RAPID, Si2, Synchronicity, the Virtual Component Exchange (VCX), and VSIA announced a successful collaboration based on QuickVC to demonstrate real-time Internet-based transactions for the exchange of virtual component (VC) information between VC providers, catalog information providers, and VC users. QuickVC will enable companies designing complex electronic products to readily find and use existing blocks of design from a broader range of sources and thus reduce time-to-market. See the press release and "QuickVC IP Exchange Over the Internet" collateral for details. Also see the DATE paper, "A Standard Internet Ready VC Exchange System."

Click here for a quick look at screen clips taken from the ECIX QuickData Toolkit Reference Implementation (i.e., customer and supplier reference software) for SoC components. This presentation shows sample queries and responses, including the QuickData XML.

Note: Si2 wishes to acknowledge NIST (National Institute of Standards and Technology) for developing with Si2 the capability for creating ECTD-format dictionary instances. This software can also be used to aid in development of other ECTD-format dictionary instances such as for QuickVC or other domains. Contact ecix-info@si2.org for additional information.