

# Electronic Component Information eXchange

## QuickData Message Specification

Version 2.0-050100

Copyright © 2000 Silicon Integration Initiative, Inc. All rights reserved worldwide.

### Introduction

This specification includes the conceptual models, semantics, and related implementation constraints that apply to all applications of QuickData information exchange.

### Requirements

The detailed user requirements and problem statements driving this specification are documented in “ECIX II Requirements”.

### Compliance

An implementation that fails to satisfy any constraint identified herein, or in the relevant normative references, with the keywords “required”, “shall”, or “shall not”, is not compliant to the specification. These keywords, along with the keywords “may”, “should”, “should not”, and “optional”, are to be interpreted as described in RFC 2119.

### Notation Conventions

This specification uses the following conventions:

1. Characters in `courier` are literal values to be used in code or messages exactly as shown.
2. Characters in `times` are not to be used literally but instead describe how values shall be constructed when used in code or messages.
3. Information contained in the following notation shall be considered non-normative:

(NOTE: This text is informative only and is not considered part of this specification.)

(EXAMPLE: This text is informative only and is not considered part of this specification.)

4. Characters bounded by double quotes have a special significance referenced as a whole in the sentence, such as a title, value, name, or other identifier.
5. An identifier bounded by angular brackets refers to a DTD element object. For example, <value> refers to the particular DTD element named “value”. If the letter “s” immediately follows the trailing right angle bracket it indicates one or more such elements.
6. An identifier bounded on the left by the percent character and on the right by the semicolon character refers to a DTD entity object. For example, %xml.att; refers to the DTD entity named “xml.att”.
7. An identifier bounded by single quotes refers to a DTD attribute object. For example, ‘NoValueReason’ refers to the DTD attribute named “NoValueReason”.

## References

### Normative

Documents identified as "Normative References" are referenced in part or in their entirety in the current document and the parts referenced shall have normative impact on compliant implementations. For each reference the edition indicated is the authority. All normative documents are subject to revision at different times and by different agencies. Consequently, application of more recent editions of such documents may affect interoperability of implementations and may have additional implications.

### Informative

Documents identified as "Informative References" are relevant to the current document, and may assist in comprehension of it, but do not have normative impact on compliant implementations.

## Terminology

The following term definitions are used in this specification. Additional terms may be defined inline, as needed, throughout the specification.

criteria specification	A request specification that includes values to be matched.
dictionary	An instance of the EC Technical Dictionary DTD.
element	A reference to some DTD object of the type “element”.
<element>	An instance of the particular QuickData DTD element that was assigned the name, “element”.
null	Element content that is empty, consisting of zero characters, not even whitespace.
product information object	Product-related data that may be in any, arbitrary format, encoding, and internal structure. (EXAMPLE: PDF datasheet or Verilog simulation model.)

request specification A request for return of a particular set of product information.

## **Conceptual Models**

The UML class diagram in Figure 1 defines the conceptual model for a QuickData transaction.

## **Message Instance Constraints and Semantics**

This section defines constraints and semantics common to all instances of a QuickData message.

### **DTD**

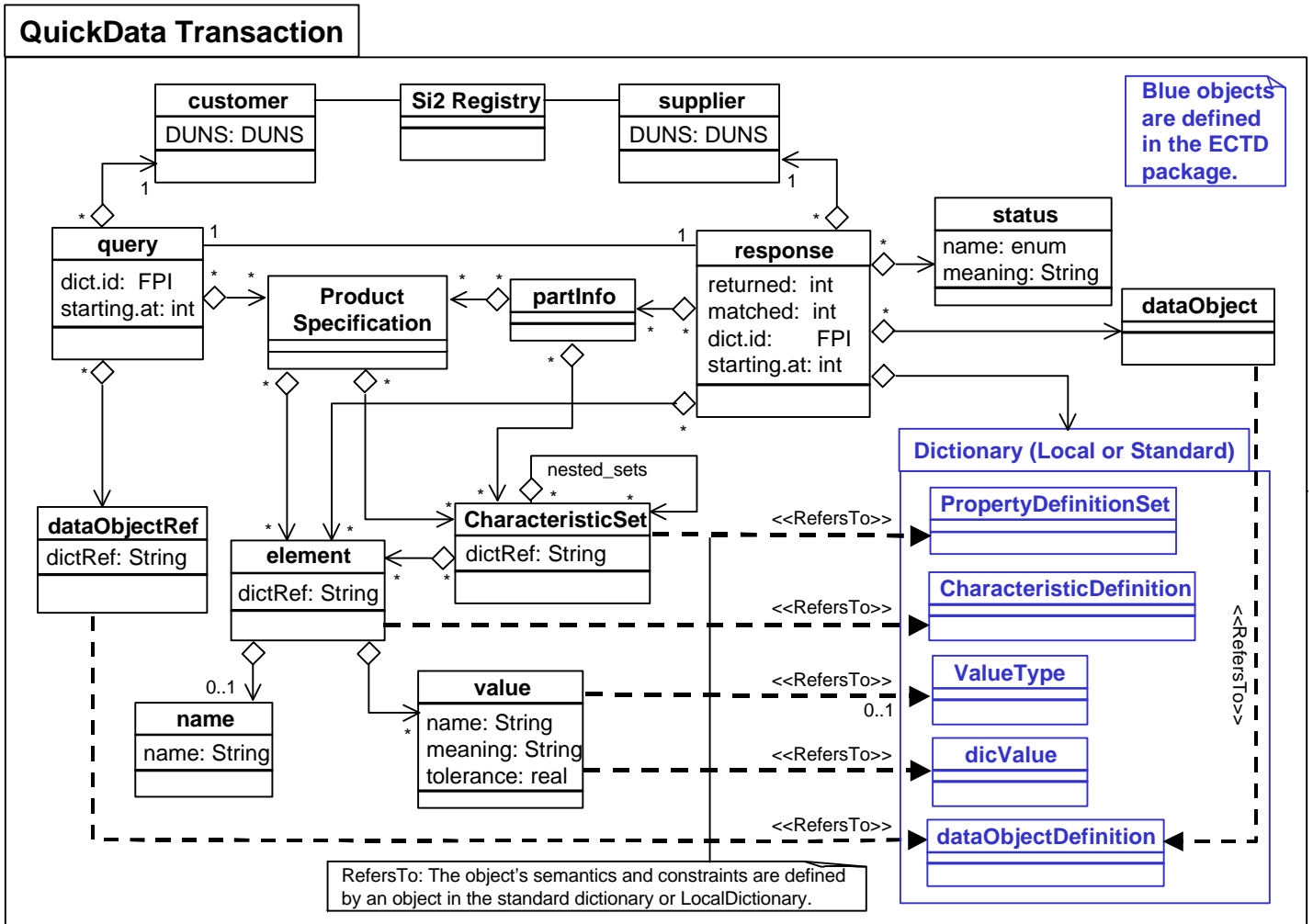
The QuickData 2.0 DTD along with the ECTD 1.0 DTD shall define the message structure.

### **Version**

An instance shall conform to the Extensible Markup Language (XML) 1.0 specification and begin with the following XML declaration:

```
<?xml version="1.0"?>
```

Figure 1: QuickData Transaction Conceptual Model



## Additional DTD Object Constraints

The QuickData DTD enforces rules such as container relationships, cardinality limitations, some presence/absence requirements, and a minimal level of data typing. However, DTD language is insufficient for the level of specification necessary to ensure machine sensibility of message content, interoperability of software that handles messages, and support for business and technical requirements of the exchange.

Therefore, the following tables impose additional constraints and semantics:

- Table 1: DTD Element Constraints and Semantics
- Table 2: DTD Attribute Constraints and Semantics
- Table 3: DTD External Entity Constraints and Semantics
- Table 4: DTD Parameter Entity Constraints and Semantics

There may be additional constraints and semantics defined by an external specification for a particular application domain.

**Table 1: DTD Element Constraints and Semantics**

Element Name	Semantics and Constraints
CharacteristicDefinition	<p>One of:</p> <ol style="list-style-type: none"> <li>1. The definition of the semantics and related information for a value reported in the current instance for which no CharacteristicDefinition exists in the standard ECTD, in which case the content shall not be empty.</li> <li>2. A stub whose 'id' attribute value matches one of the values of the 'members' attribute of a &lt;CharacteristicSet&gt; in the current instance, and also matches an 'id' attribute value of a &lt;CharacteristicDefinition&gt; in the standard ECTD, in which case the content shall be empty.</li> </ol>
CharacteristicSet	A set of characteristics and/or other CharacteristicsSets related to a product.
class	<p>The definition of the semantics and related information for product classification reported in the current instance for which no &lt;class&gt; exists in the standard ECTD.</p> <ol style="list-style-type: none"> <li>1. The content shall not be empty.</li> </ol>
data.object	<p>A request for a product information object.</p> <ol style="list-style-type: none"> <li>1. If the content matches (including case and whitespace) the &lt;preferred.name&gt; of a &lt;term&gt; entry in the standard dictionary that contains the tag:  <pre>&lt;app.specific name="object"/&gt;</pre> then the &lt;data.object&gt; constitutes a request for the type of object defined in that &lt;term&gt; entry, including all semantics, format, and encoding defined therein.</li> </ol>
dict.identifier	The unique identifier of the dictionary instance to which are compliant the semantics of all elements in the contents of the containing <query> or <response> that do not reference definitions in the local <ec.dictionary.add>.
ec.dictionary.add	A container for dictionary information referenced by the <qd> instance but not contained in the standard ECTD.
element	The association of a product information characteristic <name> with a <value>.
name	The name of a characteristic related to a product.
part.info	A container grouping information related to a component.
PropertyDefinitionSet	<p>One of:</p> <ol style="list-style-type: none"> <li>1. The definition of the semantics and related information for a PropertyDefinitionSet referenced in the current instance for which no PropertyDefinitionSet exists in the standard ECTD, in which case the content shall not be empty.</li> <li>2. A stub whose 'id' attribute value matches one of the values of the 'members' attribute of a &lt;PropertyDefinitionSet&gt; in the current instance, and also matches an 'id' attribute value of a &lt;PropertyDefinitionSet&gt; in the standard ECTD, in which case the content shall be empty.</li> </ol>
qd	The top level container for product information exchange.
query	<p>A request for product information, the content of which consists of zero or more request specifications, each of which identifies the information to be returned in a response for each product. Each request specification may be one of:</p> <ul style="list-style-type: none"> <li>• &lt;element&gt;</li> </ul>

	<ul style="list-style-type: none"> <li>• &lt;CharacteristicSet&gt;</li> <li>• &lt;data.object&gt;</li> </ul> <ol style="list-style-type: none"> <li>1. If the request specification is: <ul style="list-style-type: none"> <li>• &lt;element&gt; then the request is for the value of that element.</li> <li>• &lt;CharacteristicSet&gt; then the request is for the values for each &lt;element&gt; referenced therein, recursively for each &lt;CharacteristicSet&gt; referenced therein.</li> <li>• &lt;data.object&gt; then the request is for the actual object.</li> </ul> </li> <li>2. The presence of a &lt;value&gt; in a request specification defines a match criteria specification for only products with information matching all indicated criteria (ie, the intersection of products matching each individual criteria specification).</li> <li>3. The match criteria specification syntax for the &lt;value&gt; of any &lt;element&gt; is defined by the content of the &lt;app.specific&gt; element (in the corresponding dictionary entry) that has a 'name' attribute with a value of "query.syntax". If no such tag exists in the dictionary entry, the entire &lt;value&gt; content is to be matched exactly, except for case. The following match syntax types may be defined: <pre style="margin-left: 40px;"> &lt;app.specific name="query.syntax"&gt;EX&lt;/app.specific&gt; &lt;app.specific name="query.syntax"&gt;ML&lt;/app.specific&gt; </pre> </li> </ol> <p>All matching is case insensitive. Table 5 defines the detailed semantics for each match criteria specification syntax allowed.</p>
response	Information for zero or more products that match the criteria specified in the corresponding query.
send.addl.info	A container for one or more <data.object> requests.
status	<p>An indicator confirming success or failure of the supplier's attempt to process the &lt;query&gt;.</p> <ol style="list-style-type: none"> <li>1. If the content matches exactly the preferred name of a &lt;term&gt; entry in the standard dictionary that also includes the tag: <pre style="margin-left: 40px;"> &lt;app.specific name="status"&gt;&lt;/app.specific&gt; </pre> with a content that matches the name of the tag that is the content for the &lt;status&gt; element, then the semantics defined in that &lt;term&gt; entry shall be presumed to apply. </li> <li>2. If the content of the &lt;status&gt; is other than &lt;s.success&gt; then no &lt;part.info&gt; elements shall be included in the &lt;response&gt;.</li> </ol>
s.badquery	The <query> does not conform to all applicable specifications.
s.fail	The supplier is unable to return a useful <response> due to supplier-side system or communication problems.
s.nomatch	There is no product information matching the query criteria.
s.not.available	The product information requested may or may not exist, however, the supplier declines to return it.
s.success	Product information matching the <query> criteria is being returned in the <response>.
TargetSoftware	The definition of the semantics and related information for a <TargetSoftware>

	<p>referenced in the current instance for which no &lt;TargetSoftware&gt; entry exists in the standard ECTD.</p> <ol style="list-style-type: none"> <li>1. The content shall not be empty.</li> <li>2. The value of the 'id' attribute shall match the value of at least one 'ref.to.term' attribute in the current instance.</li> </ol>
term	<p>The definition of the semantics and related information for a &lt;term&gt; referenced in the current instance for which no &lt;term&gt; exists in the standard ECTD.</p> <ol style="list-style-type: none"> <li>1. The content shall not be empty.</li> <li>2. The value of the 'id' attribute shall match the value of at least one 'ref.to.term' attribute in the current instance.</li> </ol>
value	<p>The value associated with the containing &lt;element&gt;.</p> <ol style="list-style-type: none"> <li>1. If the defining dictionary entry has a &lt;unit&gt; then the &lt;value&gt; shall be a measurement in the unit so defined.</li> <li>2. In any &lt;element&gt; there shall be at most one &lt;value&gt; element with a 'type' attribute of a given value.</li> <li>3. In any &lt;element&gt; there shall be at most one &lt;value&gt; element with no 'type' attribute.</li> <li>4. If an &lt;element&gt; contains a &lt;value&gt; element with a 'type' attribute then that &lt;element&gt; shall not also contain a &lt;value&gt; element with no 'type' attribute.</li> </ol> <p><b>In a &lt;response&gt;:</b></p> <ol style="list-style-type: none"> <li>5. Every &lt;element&gt; shall contain at least one &lt;value&gt;.</li> </ol> <p><b>In a query:</b></p> <ol style="list-style-type: none"> <li>6. The content is a criteria match specification syntax as specified in the corresponding entry in the dictionary entry and defined in Table 5.</li> </ol>
value.codes	<p>One or more enumerated values to be added to an existing value code list.</p> <ol style="list-style-type: none"> <li>1. The value of the 'dicRef' attribute shall match the value of an 'id' attribute of a &lt;CharacteristicDefinition&gt; entry in the standard ECTD, and each child &lt;dic.value&gt; shall be considered to be part of the &lt;value.domain&gt; of that entry.</li> </ol>

**Table 2: DTD Attribute Constraints and Semantics**

Attribute Name	Semantics and Constraints
dicRef	<p>A reference to a dictionary entry.</p> <ol style="list-style-type: none"> <li>1. If used, the value of the 'dicRef' attribute shall consist of a "code part", followed by the "-" character, followed by a "majRev part". <ol style="list-style-type: none"> <li>a) The "majRev part" shall match the content of the &lt;majRev&gt; of an entry in either the dictionary identified by the &lt;dict.identifier&gt; or the &lt;ec.dictionary.add&gt; content in the current instance.</li> <li>b) The "code part" shall match the content of the &lt;code&gt; of an entry in either the dictionary identified by the &lt;dict.identifier&gt; or the &lt;ec.dictionary.add&gt; content in the current instance.</li> </ol> </li> <li>2. The entry referenced by 'dicRef' shall be a: <ol style="list-style-type: none"> <li>c) &lt;CharacteristicDefinition&gt; if 'dicRef' is an attribute of &lt;element&gt;.</li> </ol> </li> </ol>

	<p>d) &lt;PropertyDefinitionSet&gt; if 'dicRef' is an attribute of &lt;CharacteristicSet&gt;.</p> <p>e) &lt;CharacteristicDefinition&gt; if 'dicRef' is an attribute of &lt;value.code&gt;.</p>
id	<p>A unique XML identifier that may be the target of a reference from another element.</p> <ol style="list-style-type: none"> <li>1. The first 10 characters of an 'id' attribute that occurs within a &lt;ec.dictionary.add&gt; shall be the character "x" followed by the nine digits of the DUNS number of the supplier</li> </ol>
max.records	<p>A limit to the number of records returned in a &lt;response&gt;.</p> <ol style="list-style-type: none"> <li>1. If specified the value shall be a positive integer.</li> <li>2. If specified the number of &lt;part.info&gt; elements in the corresponding &lt;response&gt; shall not exceed the value of 'max.records'.</li> </ol>
matched	<p>If specified, the number of products matching the criteria specifications in the &lt;query&gt; corresponding to the current &lt;response&gt;.</p>
members	<p>The ordered set of references to the &lt;element&gt;s and/or &lt;CharacteristicSet&gt;s in the current instance that are members of the current CharacteristicSet.</p> <ol style="list-style-type: none"> <li>1. Each value shall match the value of the 'id' attribute of either a &lt;Characteristic&gt; or &lt;CharacteristicSet&gt;.</li> </ol> <p><b>In a query:</b></p> <ol style="list-style-type: none"> <li>2. Any Characteristics that appear shall be in the same relative order as the CharacteristicDefinitions referenced by the 'propDefs' attribute of the corresponding &lt;PropertySetDefinition&gt; entry.</li> </ol> <p><b>In a response:</b></p> <ol style="list-style-type: none"> <li>3. The order of the Characteristics referenced shall match the order of the CharacteristicDefinitions referenced by the propDefs of the corresponding &lt;PropertySetDefinition&gt; entry.</li> <li>4. There shall be a Characteristic included for each CharacteristicDefinition in the corresponding &lt;PropertySetDefinition&gt; entry.</li> </ol>
NoValueReason	<p>A qualifier specifying the reason for a null or empty &lt;value&gt; content.</p> <ol style="list-style-type: none"> <li>1. If the &lt;value&gt; content is non-null then the 'NoValueReason' attribute shall be null.</li> <li>2. The 'NoValueReason' attribute shall be null if a 'type' attribute is present.</li> </ol> <p><b>In a &lt;query&gt;:</b></p> <ol style="list-style-type: none"> <li>3. If the 'NoValueReason' is not specified a null &lt;value&gt; content matches any null value.</li> <li>4. Specification of a 'NoValueReason' matches any null value that is null for the indicated 'NoValueReason'.</li> </ol> <p><b>In a &lt;response&gt;:</b></p> <ol style="list-style-type: none"> <li>5. If the &lt;value&gt; content is null then the 'NoValueReason' attribute shall be non-null and shall specify the reason why the &lt;value&gt; content is null.</li> </ol>
ntol.abs	<p>A negative tolerance applicable for the current &lt;value&gt;.</p>
ntol.per	<p>A negative tolerance applicable for the current &lt;value&gt;.</p>
ptol.abs	<p>A positive tolerance applicable for the current &lt;value&gt;.</p>
ptol.per	<p>A positive tolerance applicable for the current &lt;value&gt;.</p>
qd.key	<p>An arbitrary identifier. If the 'qd.key' attribute is specified in a &lt;query&gt; then the same value shall be specified in the &lt;response&gt;.</p>

returned	The total number of <part.info> instances in the current <response>. <b>1.</b> This value shall not exceed the value of the 'matched' attribute, if any.
starting.at	The ordinal position of the first product in the ordered set of all products matching the search criteria specifications of a <query>, independent of how many are actually returned in a <response>. <b>1.</b> This value shall not exceed the value of the 'matched' attribute, if any. <b>2.</b> This value shall be greater than zero.
type	A qualifier describing the kind of measurement a <value> represents. <b>1.</b> If present, this value of the 'type' attribute shall match exactly the content of the <preferred.name> element of a <ValueType> entry in the ECTD referenced by the <dict.identifier> of the current instance, and the semantics of that <ValueType> entry shall apply to the content of the current <value>. <b>2.</b> If absent, the content of the current <value> may be interpreted by either sender or receiver of the information as any value within the domain of values that conforms to the semantics of the corresponding dictionary entry. <b>3.</b> None of the attributes 'ptol.abs', 'ptol.per', 'ntol.abs', 'ntol.per' shall be specified if value of the 'type' attribute is not "nom".
xml:lang	The language of the current element content. The semantics of this attribute are defined in section 2.12 of the "Extensible Markup Language (XML) 1.0 W3C Recommendation 10-February-1998".
xml:space	The method to be used for handling white space characters. The semantics of this attribute are defined in section 2.10 of the "Extensible Markup Language (XML) 1.0 W3C Recommendation 10-February-1998".

**Table 3: DTD External Entity Constraints and Semantics**

Entity Name	Semantics and Constraints
ECdictionary	The ECTD DTD which defines the elements in the <ec.dictionary.add>.

**Table 4: DTD Parameter Entity Constraints and Semantics**

Entity Name	Semantics and Constraints
dictionaryReference	An attribute definition for 'dicRef'.
productSpecification	An entity grouping <element>s and/or <CharacteristicSet>s that define product information.
tolerance.abs	The XML declared value for a tolerance attribute whose value is expressed as a real number to be added or subtracted from the current <value>.
tolerance.per	The XML declared value for a tolerance attribute whose value is expressed as a real number that is a percent of the current <value>.
xml.atts	Attribute definitions for the XML declaration attributes, 'xml:space' and 'xml:lang'.

**Table 5: Criteria Match Specification Syntax**

Content	Name	Query Syntax Description
<i>NONE:</i>	single	The <value> content is a single value. A query shall

<i>tag is absent</i>	value	match components whose value for the corresponding element field matches exactly (except for case) the value specified in the query. This is the default query syntax rule if no <code>&lt;app.specific name="query.syntax"&gt;</code> tag exists.		
ML	Match List	The <code>&lt;value&gt;</code> content is a blank-separated list of strings. A query shall match components whose values for the corresponding element match any one of the blank-separated strings in the list, but the values must match exactly (except for case). Strings to be matched that include blanks shall be surrounded in double quotes.		
EX	Keyword Expression	The <code>&lt;value&gt;</code> content is one or more substrings to be matched, along with optional expression operators:		
		includes	+ & AND	match includes both left and right substring
		excludes	- NOT	match does not include
		or	 OR	match includes either left or right substring
		grouping	( )	group boolean sub expressions
		phrase	" "	match must include the entire phrase exactly (except for case) that is bounded by the quotes (useful with embedded blanks or characters that are also operators)
A match occurs if the substring appears anywhere in the data to be searched. All matches are case insensitive. The expression is evaluated from left to right, except that an expression inside grouping characters is evaluated before continuing with left-to-right evaluation. Blanks (not appearing inside quotes) separate substrings with an implied OR operation, unless there is an explicit operator before or after, in which case the blanks are ignored. Note: There are no explicit "wild cards" that can be imbedded in the middle of a string.				

## Normative References

### Requirements

ECIX II Requirements: [requirements.html](#).

## QuickData Specifications

QuickData 2.0 DTD: [DTD\QD\qd.dtd](#).

ECTD 1.0 DTD: [DTD\ECTD\ec-new.dtd](#).

## Other Specifications

Electronic Components Technical Dictionary Specification: [ectd.doc](#).

Si2 Registry Specification: [si2\\_registry.doc](#).

Extensible Markup Language (XML) 1.0 W3C Recommendation 10-February-1998:  
<http://www.w3.org/TR/REC-xml>.

RFC 2119, “Key words for use in RFCs to Indicate Requirement Levels”:  
<http://andrew2.andrew.cmu.edu/rfc/rfc2119.html>

## Informative References

OMG Unified Modeling Language Specification, Version 1.3, June 1999: <http://www.omg.org/cgi-bin/doc?ad/99-06-08>