Reviews

ThermoML†—An XML-Based Approach for Storage and Exchange of Experimental and Critically Evaluated Thermophysical and Thermochemical Property Data. 3. Critically Evaluated Data, Predicted Data, and Equation Representation‡

Michael Frenkel,* Robert D. Chirico, and Vladimir V. Diky

Thermodynamics Research Center (TRC), Physical and Chemical Properties Division, National Institute of Standards and Technology (NIST), 325 Broadway, Boulder, Colorado 80305-3328

Kenneth N. Marsh

Department of Chemical and Process Engineering, University of Canterbury, Private Bag 4800, Christchurch, New Zealand

John H. Dymond

Chemistry Department, University of Glasgow, Glasgow G12 8QQ, UK

William A. Wakeham

School of Engineering Sciences, University of Southampton, Highfield, Southampton SO17 1BJ, UK

ThermoML is an XML-based approach for storage and exchange of experimental and critically evaluated thermophysical and thermochemical property data. Extensions to the ThermoML schema for the representation of predicted data, critically evaluated data, and data expressed as equations are described. The present role of ThermoML in global data submission and dissemination is discussed with particular emphasis on cooperation between major journals in the field and TRC at NIST. The text of several data files illustrating the new extensions is provided as Supporting Information together with the complete updated ThermoML schema text.

Background

This is the third paper in the series describing ThermoML, an XML-based approach for storage and exchange of thermophysical and thermochemical property data.[1,2] The principles, scope, and description of the basic structural elements of ThermoML were discussed in the first paper in this series.[1] ThermoML covers essentially all experimentally determined thermodynamic and transport property data (more than 120 properties) for pure compounds, multicomponent mixtures, and chemical reactions (including change-of-state and equilibrium) with a primary focus on molecular compounds. Properties of polymers, radicals, and some properties of ionic systems are not represented at present. Expansion of ThermoML to cover these systems is under development. Representation of quantities for the expression of uncertainty was discussed in the second paper of the series.[2] All quantities related to the expression of uncertainty within ThermoML conform to the Guide to the Expression of Uncertainty in Measurement, *ISO (International Organization for Standardization), October, 1993*.[3] These ISO recommendations were adopted with minor editorial changes as the *U. S. Guide to the Expression of Uncertainty in Measurement*, commonly referred to as the GUM.[4] The recommendations have been summarized in *Guidelines for the Evaluation and Expression of Uncertainty in NIST Measurement Results*,[5] which is available via free download from the Internet (http://physics.nist.gov/cuu/).

In this paper, we describe the expansion of ThermoML for coverage of critically evaluated data, predicted data, and data represented as equations. The complete ThermoML schema is available on the Internet (http://www.trc.nist.gov/ThermoML.xsd). ("xsd" is the file extension for an xml schema.)

Basic Definitions

The framework of ThermoML[1] was developed to provide representation of metadata and numerical data for thermophysical and thermochemical properties obtained from experiments as well as some property data derived directly from these experimental results. Although it was feasible to include *predicted* and *critically evaluated* data in the existing ThermoML schema by simply considering these

† "ThermoML" is the reserved namespace for the XML-based IUPAC standard for experimental and critically evaluated thermodynamic property data storage and capture (http://www.iupac.org/namespaces/ThermoML/).

* To whom correspondence may be addressed. E-mail: frenkel@boulder.nist.gov.

10.1021/je049890e CCC: $27.50
Published on Web 04/17/2004

as "other methods" of obtaining property data, it is our experience that clear distinctions between these data types and *experimental* data are often of prime importance to the academic researcher, data evaluator, process engineer, or other end user. Consequently, it was decided to include *predicted* and *critically evaluated* data explicitly in the ThermoML schema. To cover these types of thermophysical and thermochemical property data, it is necessary to establish some definitions. We are not aware of any such definitions in the literature that might reflect a consensus of the scientific community. The definitions provided here are not intended in any way to serve as a rigorous guide (or "standard") for distinguishing various types of property data but rather to provide clarification in the interpretation and use of the structural elements of ThermoML.

True Data (Hypothetical). *True* data are exact property values for a system of defined chemical composition in a specified state. These data have the following characteristics. They are (1) unique and permanent, (2) independent of any experiment or sample, and (3) a hypothetical concept with no known values. Because they are hypothetical, *true* values are not represented in ThermoML. The property types that are represented in ThermoML (experimental, predicted, and critically evaluated) may be considered approximations to the true values. The difference between the represented values and true values is defined as the *error*. The *error* is never known; however, it is given that it is never zero. Like *true* values, *errors* are not represented in ThermoML. A measure of the quality or confidence in an experimental, predicted, or critically evaluated value is expressed in terms of the "uncertainty,"[3] which is a range of values believed to include the *true* value with a certain probability. Representation of uncertainties in ThermoML was addressed in the second paper of this series.[2] All data types can and should have associated uncertainty estimates.

Experimental Data. Experimental data are defined in the context of ThermoML as those obtained as the result of a particular experiment on a particular sample by a particular investigator. Representation of experimental data was covered fully in the first paper in this series.[1] The feature that distinguishes *experimental* data from *predicted* and *critically evaluated* data is use of a chemical sample including characterization of its origin and purity.

Predicted Data. Predicted data are defined here as those obtained through application of a predictive model or method such as a particular molecular dynamics, corresponding states, group contribution method, etc. (A more complete enumeration list is included within the description of the new ThermoML elements later in this paper.) Clearly, there is no sample associated with this type of property data.

Critically Evaluated Data. Like predicted data, there is no sample involved with critically evaluated data. The feature that distinguishes *critically evaluated* data from *predicted* data is the involvement of the judgment of a data evaluator or evaluation system. Critically evaluated data are recommended property values generated through assessment of available *experimental* and *predicted* data or both. This definition makes no distinction between values derived by traditional "static" data-evaluation methods and proposed "dynamic" methods for critical evaluation of data "to order" using comprehensive (and dynamically populated) data storage facilities (containing all pertinent information available to date) and autonomous software expert systems.[6,7] In addition to new elements for specification of property data as *critically evaluated*, subelements will be described (see Extensions to the ThermoML schema), which allow further distinction based upon the degree of inter-property consistency enforced in the evaluation process.

Derived Data. *Derived* data can be defined as property values calculated by mathematical operations from other data, possibly including *experimental*, *predicted*, and *critically evaluated* data. For this reason *derived* data are not represented as a separate data type in ThermoML. Some derived properties, such as the absolute entropy at temperature T, derived from the results of adiabatic calorimetry from near 0 K to T K for a particular sample, are derived by mathematical manipulation of the primary experimental data only, and consequently, are predominantly *experimental* in nature. Others, such as enthalpies of vaporization derived from *experimental* vapor pressures with the Clapeyron equation, will necessarily include *critically evaluated* volumetric property data for the gaseous and condensed phases. Whether these values are represented as *experimental* or *critically evaluated* is left to the best judgment of the ThermoML user (the data compiler). Although the focus of ThermoML is on properties determined either by direct experimental measurement or through application of some predictive model, ThermoML does cover a broad range of key derived property data such as azeotropic properties, Henry's Law constants, virial coefficients (for pure compounds and mixtures), activities and activity coefficients, fugacities and fugacity coefficients, and standard properties derived from high-precision adiabatic heat-capacity calorimetry. The complete list of properties available currently is provided in the first paper of this series[1] and is available on the Web.[8]

Equation Representation of Property Data. Until now, elements of ThermoML allowed representation of property values only as discrete data points.[1,2] In the present paper, extensions are described to allow representation of property data as mathematical equations. The modular nature of XML is exploited here and allows published MathML formats[9] to be used in conjunction with ThermoML and eliminates the need to develop here the complex structures needed to represent mathematical expressions. The mathematical markup language (MathML) is a low-level specification for describing mathematics as a basis for machine-to-machine communication in terms of both content (mathematical meaning) and presentation (format). MathML (version 2.0) is a W3C (World Wide Web Consortium)[10] recommendation and was released February 21, 2001. It will be shown that equations can be defined by any user of ThermoML through use of the new *ThermoML−EquationDefinition* schema. By linking the *ThermoML−EquationDefinition* schema to MathML, it is possible to take advantage of the full scope of elements developed for MathML in construction of the equation definition. Schema extensions to ThermoML for equation representation provide the elements necessary for defined storage of the various equation components required for the selected specific equation definition. The nature or scope of the equations is not restricted in any way. A more complete description of the implementation of this approach is provided later together with several examples.

Some equation templates formulated with MathML are provided here for a variety of common equation types, such as the Antoine equation for vapor pressures or a polynomial equation in terms of different variable powers commonly used for representation of a wide variety of properties such as heat capacities or densities at saturation or constant pressure over relatively short temperature intervals. No

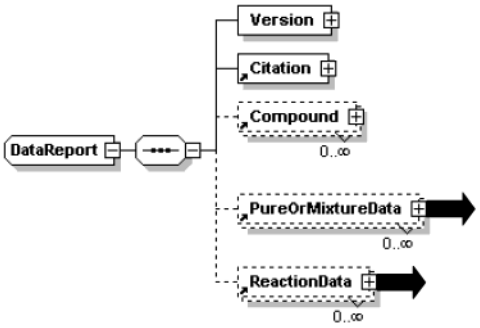

Figure 1. Major components of ThermoML.

attempt has been made (or will be made) to make this collection of templates comprehensive, because the variety of possible equation representations is essentially infinite. The selection of the initial collection of equations was based on their widespread use in existing data collections for chemical engineering. The provided templates serve as a convenience to ThermoML users who require the given specific formulations and also as examples for their general methods of construction. The structure of MathML syntax is not described in this paper; however, full descriptions are readily available.[9]

Units. By design, there is only one unit selected for each property covered by ThermoML. These units are SI; however, for a number of properties, the selected units are multiples of SI units to ease interpretation of numerical values. Unit tagging is explicit in a ThermoML file as part of each property name, thus minimizing the possibility of unit misinterpretation. As a natural extension to this approach, the units for values of uncertainty are in most cases the same as those of the quantity whose uncertainty they represent. A few uncertainties are represented as percentages. These are indicated clearly in the names and element descriptions. It is recognized that some users of ThermoML might prefer to express their equations in units other than those defined within ThermoML, and this option is provided. All formulations provided in the ThermoML library of equation definitions conform to the SI unit conventions of ThermoML.

Tagging. The names or "tags" for the elements of the schema include special characters related to the type of information to be stored. A name beginning with "e" designates an *enumeration* element (i.e., values selected from a predefined list), "s" designates a *string* element, "n" specifies a *numerical* element (integer or floating), and "url" designates a URL or *uniform resource locator*, which is the World Wide Web address of a site on the Internet. Elements identified by dotted boxes in a figure are optional, and those in solid-lined boxes are mandatory. Complex elements are defined as those that have internal structure (i.e., "subelements"). Complex elements illustrated without their internal structure are identified with "+." Complex elements with internal structure displayed are identified with "−." Multiple elements of the same type are identified by lower and upper limits listed below the relevant boxes in the figures. Within ThermoML, the only limits used are "0..∞" for optional elements and "1..∞" for mandatory elements.

Implementation of Structural Elements Related to Predicted and Critically Evaluated Data in ThermoML

ThermoML consists of four major blocks, which are shown in Figure 1 together with the element **Version** [complex], which specifies the ThermoML version number. The four major blocks were described previously.[1]

The general locations of extensions described in this paper for predicted and critically evaluated data are indicated in Figures 1−3. Detailed schema figures and the text of ThermoML were created with the software package XML SPY.[11] (The trade name is provided only to specify the procedure adequately and does not imply endorsement by the National Institute of Standards and Technology. Similar products by other manufacturers may be found to work as well or better.) The four major blocks are:

(1) *Citation* (description of the source of the data),

(2) *Compound* (characterization of the chemical system),

(3) *PureOrMixtureData* (meta- and numerical data for a pure compound or multicomponent mixture),

(4) *ReactionData* (meta- and numerical data for a chemical reaction with a change of state or in chemical equilibrium).

The extensions described in this paper for predicted and critically evaluated data are exclusively in blocks 3 and 4 and are associated with metadata for the properties only, as indicated in Figures 2 and 3. The **Property** [complex] is a major mandatory element in both the *PureOrMixtureData* and *ReactionData* blocks. Among other subelements, **Property** is characterized by **PropertyMethodID** [complex], which identifies the property and the method used. **PropertyMethodID** includes **PropertyGroup** [complex] in both the *PureOrMixtureData* and *ReactionData* blocks, as shown in Figures 2 and 3. The **PropertyGroup** [complex] element in the *PureOrMixtureData* block is represented by 10 property groups (Figure 4). The names of the property groups are designed to reflect the nature of the specific properties included in the group, as described in the first paper of this series.[1] For example, **VaporPBoilingTAzeotropTandP** [complex] is the property group that includes *vapor pressure or sublimation pressure, normal boiling temperature, boiling temperature at pressure P, azeotropic temperature, and azeotropic pressure*. Thermophysical properties are divided into a total of 10 groups to simplify the property-selection process for the ThermoML user. Similarly, the **PropertyGroup** [complex] element in *ReactionData* block is represented by two property groups: **EnthalpyInternalEnergyOfReaction** [complex] and **ReactionEquilibriumProp** [complex], as shown in Figure 5.

Extensions to the ThermoML Schema for Predicted and Critically Evaluated Data

Extensions related to *predicted* and c*ritically evaluated* data are part of the internal structure of each property group in both the *PureOrMixtureData* and *ReactionData* blocks. Previously, the structure of each property group consisted of **ePropName** [enumeration], **eMethodName** [enumeration], and **sMethodName** [string] elements. Enumeration lists of property names and associated *experimental* methods characterize the elements **ePropName** and **eMethodName** within each property group. If the method used is not listed, the string element **sMethodName** can be used to identify it. To incorporate the extensions for *predicted* and *critically evaluated* data, new complex subelements, **Prediction** [complex] and **CriticalEvaluation** [complex] are added (Figures 4 and 5). Therefore, each property group is now characterized with **ePropName** [enumeration] identifying the property and one of four elements characterizing the method used to determine it. Two of the four elements, **eMethodName** [enumeration] and **sMethodName** [string], serve to iden-

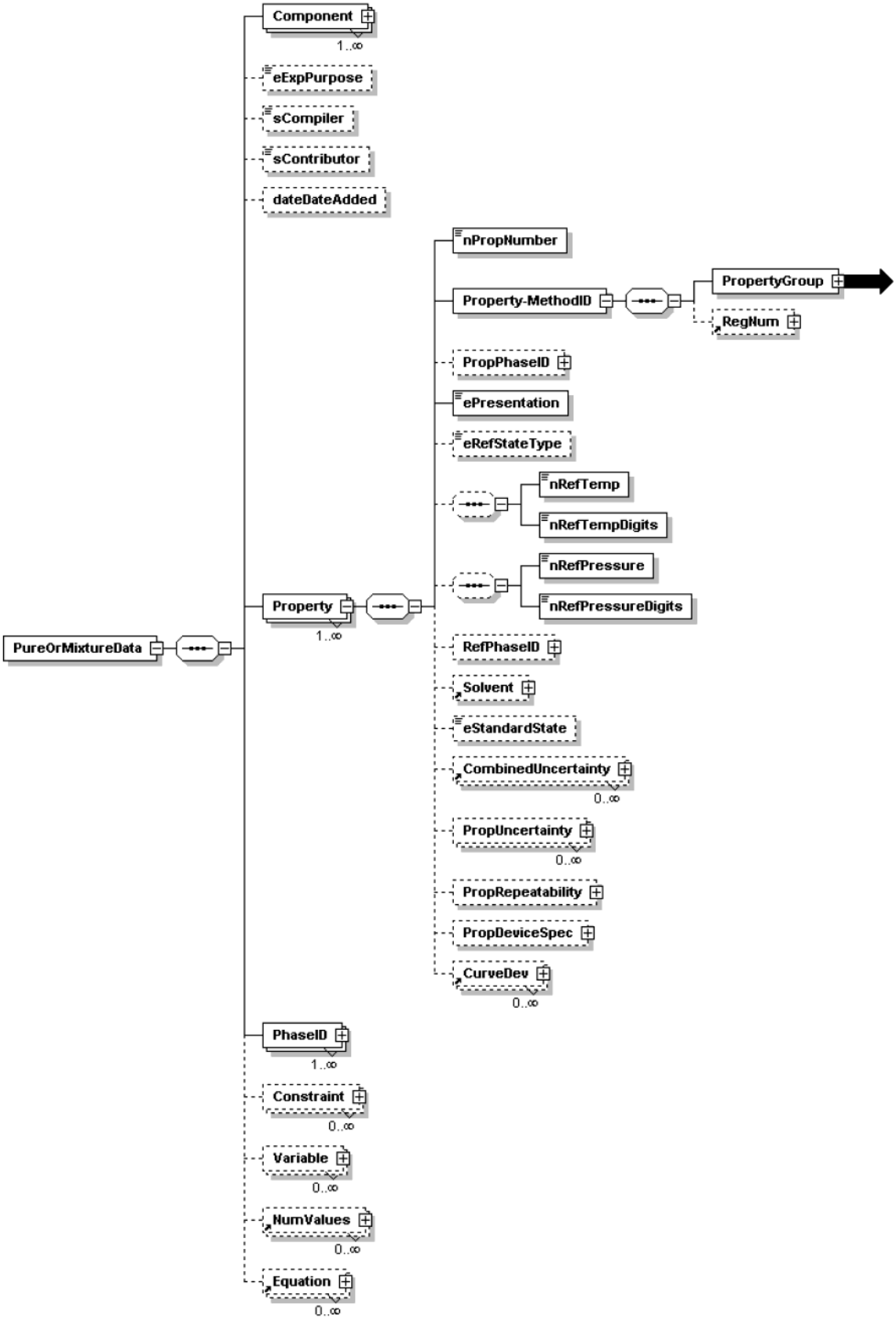

Figure 2. Structure of the **Property** element in the *PureOrMixtureData* block. The arrow indicates the location of extensions to the schema for representation of predicted and critically evaluated data.

tify experimental methods. The element **Prediction** [complex] is added to describe the prediction method used, while the element **CriticalEvaluation** [complex] is added to describe the critical evaluation method.

Prediction [complex] contains one mandatory element **ePredictionType** [enumeration] and three optional elements: **sPredictionMethodName** [string], **sPredictionMethodDescription** [string], and

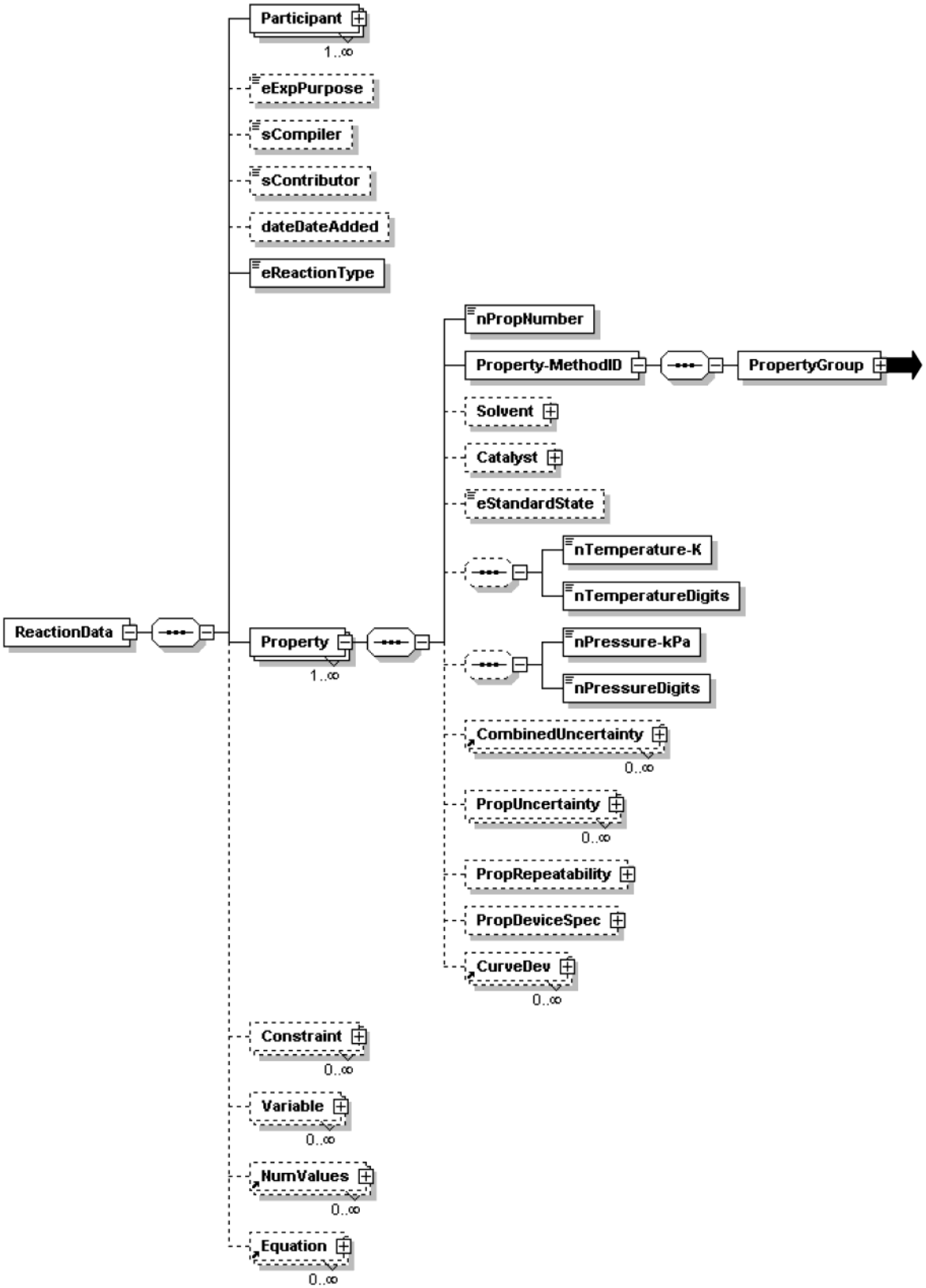

Figure 3. Structure of the **Property** element in the *ReactionData* block. The arrow indicates the location of extensions to the schema for representation of predicted and critically evaluated data.

PredictionMethodRef [complex] (Figure 6). The **ePredictionType** [enumeration] element allows selection of one major type of prediction method (ab initio, molecular dynamics, semiempirical quantum methods, statistical mechanics, corresponding states, correlation, and group contribution). The element **sPredictionMethodName** [string] serves to identify the name of the prediction method. This is particularly helpful in identifying the method, if this method name is well established. **sPredictionMethodDescription** [string] could be used to describe the principal features of the method, its limitations, assumptions, etc. If the method used for the

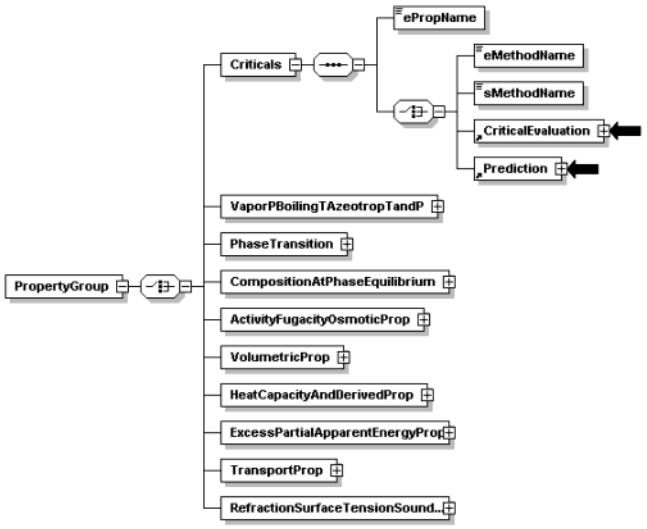

Figure 4. Structure of the **PropertyGroup** element in the *PureOrMixtureData* block. The arrows indicate new elements within one property group for representation of predicted and critically evaluated data described in the text. Analogous extensions within the other nine property groups are not shown.

prediction has been described in the literature, the element **PredictionMethodRef** [complex] can be used to identify the original reference(s). **PredictionMethodRef** [complex] has the same structure as the major element **Citation** listed in Figure 1. A wide variety of citation types are supported, as was shown in Figure 2 of the first paper in this series.[1]

The element **CriticalEvaluation** [complex] allows selection of one of three elements: **SingleProp** [complex], **MultiProp** [complex], or **EquationOfState** [complex] (Figure 7). **SingleProp** [complex] is designed to identify a critical evaluation method based on analysis for a single property only without consideration of interproperty consistency. The property is identified in the **ePropName** [enumeration] element (Figures 4 and 5). **SingleProp** [complex] contains **sEvalSinglePropDescription** [string] and the multiple element **EvalSinglePropRef** [complex]. For example, the method used for critical evaluation of density data along the saturation line published recently[12] could be described in **sEvalSinglePropDescription** [string] as, "*Weighted least-squares fitting to a 4th-order polynomial at low temperatures and weighted-least-squares fitting to the Guggenheim equation at higher temperatures. The weights are based on the uncertainties of the selected experimental data*." Information about ref 12 could be incorporated into the element **EvalSinglePropRef** [complex], which has the same structure as **Citation** and allows full specification of the reference.

MultiProp [complex] allows identification of a critical evaluation method as enforcing mutual consistency for a limited number of related properties. The structure of **MultiProp** [complex] is similar to the structure of **SingleProp** [complex] with **sEvalMultiPropDescription** [string] analogous to **sEvalSinglePropDescription** [string], and **EvalMultiPropRef** [complex] is similar to **EvalSinglePropRef** [complex]. However, **MultiProp** [complex] has one additional subelement, **sEvalMultiPropList** [string], in comparison with **SingleProp** [complex]. **sEvalMultiPropList** [string] identifies the properties involved in the multiple property critical evaluation. For example, experimental saturated vapor pressure data are often evaluated together with calorimetric enthalpy-of-vaporization data, experimental heat capacity data in the liquid state, and heat capacity data in gas phase (commonly calculated by the method of statistical mechanics). In this case, the list of properties should be provided in the element **sEvalMultiPropList** [string].

Finally, **EquationOfState** [complex] is designed to allow identification of a particular equation of state used to enforce general thermodynamic consistency. The elements **sEvalEOSDescription** [string] and **EvalEOSRef** [complex] are completely analogous to elements in **SingleProp** [complex] and **MultiProp** [complex]. **EquationOfState** [complex] also includes the element **sEvalEOSName** [string] to identify the name of the equation of state used (if applicable).

There are no special extensions to ThermoML required for specification of uncertainty information for predicted and critically evaluated data. Representation of uncertainties was discussed fully in the second paper of this series.[2]

General Implementation of Equation Representations in ThermoML

To understand the need for several key new elements in the ThermoML schema, it is necessary to understand the relationships among the following five items: the ThermoML schema (*ThermoML.xsd*), a ThermoML data file that includes an equation representation (an XML file), the ThermoML Equation Definition schema (*ThermoMLEquation.xsd*), a ThermoML equation definition file (an XML file), and MathML (*mathml2.xsd*). A general picture is provided in Figure 8. (The ThermoML schema is presently located on the NIST/TRC Web site. Presently, ThermoML is being developed as the core of a new IUPAC standard, "XML-based IUPAC Standard for Experimental and Critically Evaluated Thermodynamic Property Data Storage and Capture." When the standard is completed, the location is expected to change to an IUPAC Web site.)

The boxes on the right-hand side of Figure 8 represent a ThermoML data file (the lower box) and the ThermoML schema (the upper box) used to validate it. The ThermoML data file includes the identities of all variables, fitted parameters, and constants that are required for a particular equation representation but does not contain any

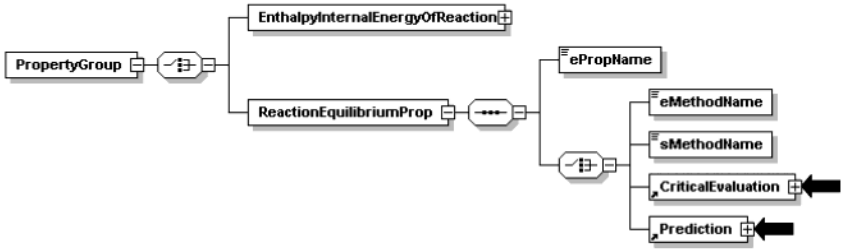

Figure 5. Structure of the **PropertyGroup** element in the *ReactionData* block. The arrows indicate new elements for representation of predicted and critically evaluated data described in the text. Analogous extensions within the **EnthalpyInternalEnergyOfReaction** group are not shown.

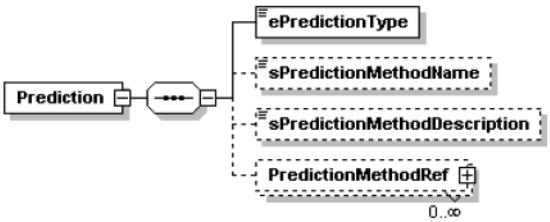

Figure 6. Structure of the **Prediction** element.

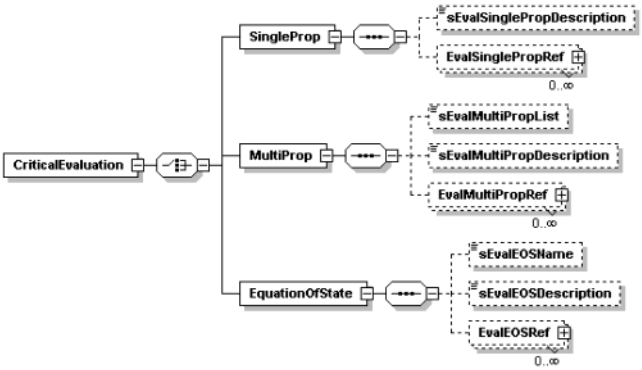

Figure 7. Structure of the **CriticalEvaluation** element.

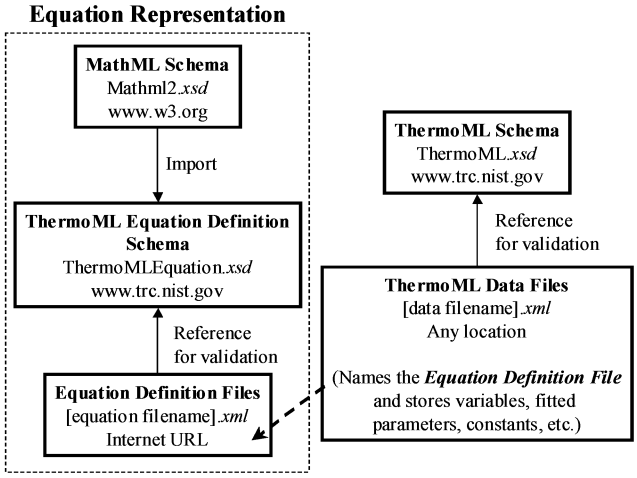

Figure 8. Components of equation representation with ThermoML. The components within the dotted line are required to give meaning to the equation information stored in the ThermoML data files (shown in the lower right). Each component is listed with its file name and/or type ("xsd" identifies a schema and "xml" identifies an xml file) and general location.

mathematical expressions. One element of the ThermoML data file is a URL used to specify the Internet location (URL) of the full equation definition. The three boxes on the left-hand side of Figure 8 (within the dotted line) are required to give meaning to the equation information stored in the ThermoML data file (lower right). The ThermoMLEquation schema (middle left of Figure 8) is designed for storage and exchange of equation definitions with full mathematical content included through importation of the MathML schema. ThermoMLEquation is a general schema for the definition of any type of equation for representation of thermophysical and thermochemical properties. An equation definition file (lower left of Figure 8) is created for the definition of a particular equation (e.g., the Wagner equation for vapor pressures). Care must be taken by the ThermoML file creator to ensure that the identities of the property, variables, constraints, and equation parameters and constants are correctly linked in the ThermoML file and the ThermoMLEquation file. Linking is accomplished through equation symbols and indexes, as described later.

At present, the MathML schema is used in conjunction with the ThermoMLEquation schema strictly for communication of mathematical content, i.e., to communicate mathematical meaning. However, MathML can also be used for transfer of information concerning presentation, making it possible to include thermodynamic property symbols, which are in full accord with IUPAC recommendations provided in the Green Book.[13]

The meanings of new elements defined here for equation representation with ThermoML correspond to the various components of a mathematical expression for relating thermophysical and thermochemical properties. These components are *properties*, *constraints*, *variables*, *parameters*, *constants*, *indexes*, and the *covariance matrix*. Properties, variables, and constraints are defined within ThermoML,[1] and further discussion is not needed here. Within the equation representations, these quantities are treated equally as functions of state and can appear anywhere in the equation representations. This avoids difficulties associated with rearrangement of certain equations to specify a single property in terms of all other quantities. (The virial equation of state is an example of such an equation.) The ranges within which an equation is valid for the property, variable, and constraint are defined within ThermoML. *Parameters* are adjustable or "fitted" quantities, which are typically varied to optimize agreement between experimental and calculated values. *Constants* are other numerical values in an equation that are not adjustable. Many constants can be specified uniquely in an equation, but for some equations, a more general formulation is more convenient. For example, the Wagner equation is often used to represent vapor pressure results of high quality. A commonly used form of the Wagner equation is

$$\ln(p/p_c) = (1/T_r)\{A_1(1 - T_r) + A_2(1 - T_r)^{1.5} + A_3(1 - T_r)^{2.5} + A_4(1 - T_r)^5\} \quad (1)$$

where T_r is the reduced temperature T/T_c, p is the pressure, and T_c and p_c are, respectively, the critical temperature and critical pressure. The quantities A_1, A_2, A_3, an A_4 are the adjustable equation *parameters*, and the exponents (1.5, 2.5, and 5) are part of the equation definition. Various forms of the Wagner equation are used by researchers with alternative exponent values, such as (1.5, 3, 6) or (2, 3, 4). Consequently, it is convenient to define the Wagner equation in more general terms

$$\ln(p/p_c) = (1/T_r)\sum_{i=1}^{nPar} A_i (1 - T_r)^{n_i} \quad (2)$$

with *nPar* terms where the exponents (n_i) can be defined within ThermoML along with the *parameters* A_i.

The *covariance* is defined for each pair of adjusted *parameters*. If covariance values are not provided for any parameters, ThermoML does not provide the means to store information concerning which parameters were adjusted in the fitting procedure and which were not. Such information is beyond the scope of ThermoML. Covariance values are never associated with *constants* because constants are not adjusted, by definition. *Indexes* are used to map a particular *property*, *constraint*, *variable*, *parameter*, or *constant* to a summation or product index in a defined equation. For example, the parameter *A* in Wagner eq 1 has *indexes* 1−4. There are two important conventions related to indexes in ThermoML. Indexes in ThermoML

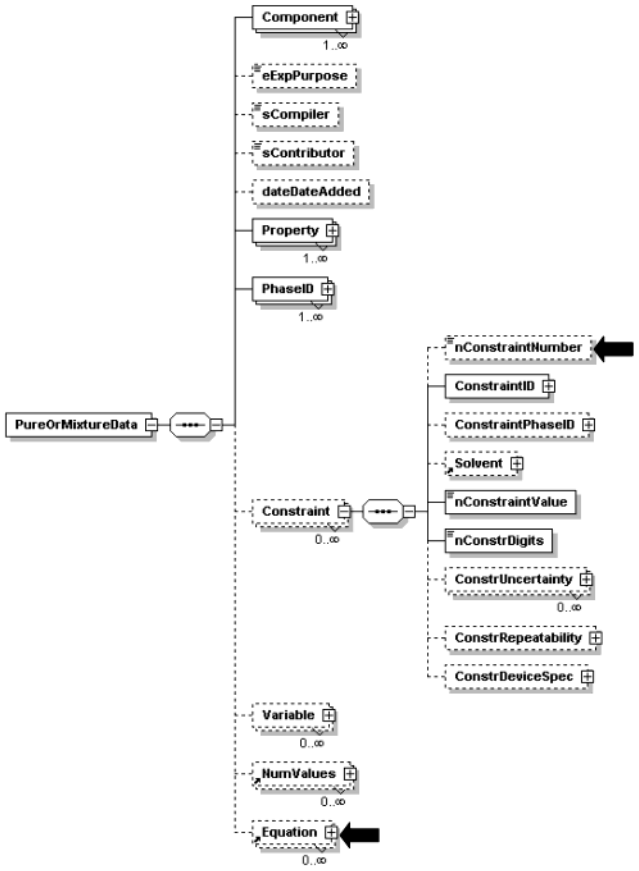

Figure 9. Structure of the *PureOrMixtureData* block and the **Constraint** element. The arrows indicate new elements for equation representation.

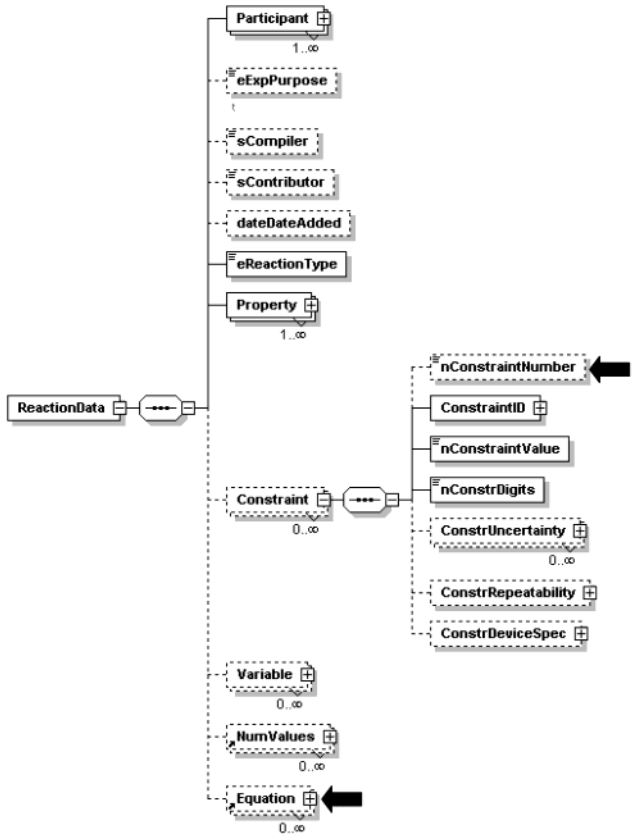

Figure 10. Structure of the *ReactionData* block and the **Constraint** element. The arrows indicate new elements for equation representation.

equations are integers starting from 1, and missing parameter values and constant values are substituted by zeros.

Extensions to the ThermoML Schema for Equation Representation of Property Data in ThermoML

The locations of extensions described in this paper for equation representation of property data are indicated in Figures 9 and 10. Subelements of the new element **Equation** [complex] are identical for the *PureOrMixtureData* and *ReactionData* blocks. Consequently, only one description of the subelements of **Equation** [complex] is necessary.

One additional element, **nConstraintNumber** [numerical, integer] is now added to the element **Constraint** [complex] to ease equation representation. This addition is made in both the *PureOrMixtureData* and *ReactionData* blocks, as shown in Figures 9 and 10. If multiple constraints are involved in a particular representation, this additional element is used to ensure that the constraints are properly identified and ordered for linking between ThermoML and the ThermoMLEquation representations. The analogous elements **nVarNumber** [numerical, integer] and **nPropNumber** [numerical, integer] exist already within ThermoML.[1]

The structure of the new element **Equation** [complex] for the *PureOrMixtureData* and *ReactionData* blocks, is shown in Figure 11. Complex subelements of **Equation** [complex] are shown in Figures 12−17. The first pair of elements within **Equation** [complex], **eEqName** [enumeration], and **sEqName** [string] allow specification of an equation name. The element **eEqName** [enumeration]

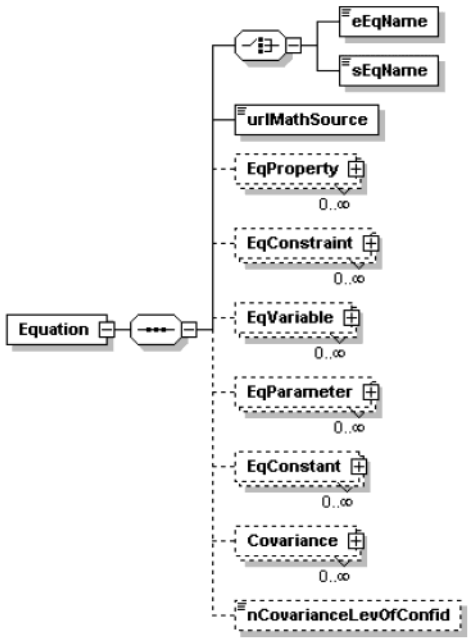

Figure 11. Structure of the **Equation** element for equation representation. This element occurs in the *PureOrMixtureData* block and the *ReactionData* block.

allows selection from a list of the equation names provided within the ThermoML library of equation representations in ThermoMLEquation format. This list will grow as new representations are added. The element **sEqName** [string] should be used to name an equation that is not part of the ThermoML library of equations. The element **urlMathSource** [Internet address] is the location on the

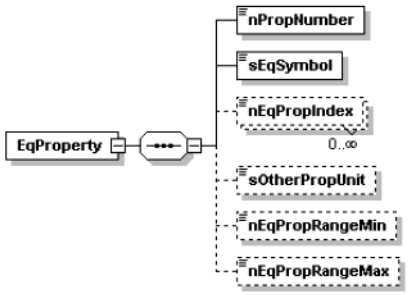

Figure 12. Structure of the **EqProperty** element for equation representation. **EqProperty** is a subelement of **Equation** [complex].

Internet where the XML representation of the specified equation is stored. The remaining complex elements of **Equation** [complex]: **EqProperty** [complex], **EqConstraint** [complex], **EqVariable** [complex], **EqParameter** [complex], and **EqConstant** [complex] are used to represent the components of an equation. *Indexes* are represented within these complex elements to allow vector or matrix representation, such as that shown earlier for the parameters of the Wagner equation. **Covariance** [complex] stores the covariance for each of the equation parameter pairs, and **nCovarianceLevOfConfid** [numerical, floating] stores the level of confidence[2] associated with uncertainties calculated with the covariance values.

The names used for equations within the ThermoML library of equation representations follow the convention: "*ThermoML.(equation name).(property)*". An example of this convention is "*ThermoML.Wagner25Linear.VaporPressure*". It is recommended that other ThermoMLEquation representations created for use with ThermoML conform to this format, but this is not required. Duplicate equation names are not a problem because uniqueness is enforced through the particular **urlMathSource** [Internet address] specified for the equation definition and not through the equation name.

The structure of the subelement **EqProperty** [complex] within **Equation** [complex] is shown in Figure 12. Within ThermoML, it is possible to include any number of properties within the elements **PureOrMixtureData** [complex] and **ReactionData** [complex] for a given chemical sample, mixture, or chemical reaction, as was shown in Figures 8 and 12 of the first paper in this series.[1] **nPropNumber** [numerical, integer] within **EqProperty** [complex] provides the mechanism to identify a specific property. **sEqSymbol** [string] is used to map a symbol used in the equation definition (a ThermoMLEquation file) to a particular property defined in the ThermoML file. **nEqPropIndex** [numerical, integer] is used to map the property to a particular index in the equation definition. **sOtherPropUnit** [string] allows the user to define any unit preferred for the property. Because there are an infinite number of possible units, no attempt was made to create an enumeration list. Provision of the **sOtherPropUnit** [string] element was included to facilitate, for example, intracompany or interpersonal communications. All equations provided in the NIST/TRC library of representations include only SI units defined in the original ThermoML specifications.[1] **nEqPropRangeMin** [numerical, floating] and **nEqPropRangeMax** [numerical, floating] define the range within which the equation is valid for the particular property.

The structure of the subelement **EqConstraint** [complex] within **Equation** [complex] is shown in Figure 13. The six subelements of **EqConstraint** [complex]: **nConstraintNumber** [numerical, integer], **sEqSymbol** [string], **nEqConstraintIndex** [numerical, integer], **sOtherConstraintUnit** [string], **nEqConstraintRangeMin** [numerical, floating], and **nEqConstraintRangeMin** [numerical, floating], are analogous to those of **EqProperty** [complex] described above. The term *constraint* used in the context of **EqConstraint** [complex] refers to a *constraint* defined as an immediate subelement of the *PureOrMixtureData* or *ReactionData* blocks. It is not necessarily a constraint for the equation. For example, an equation for which pressure is not constrained might be used to represent an experimental data set in which it is.

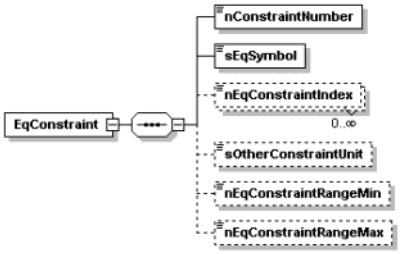

Figure 13. Structure of the **EqConstraint** element for equation representation. **EqConstraint** is a subelement of **Equation** [complex].

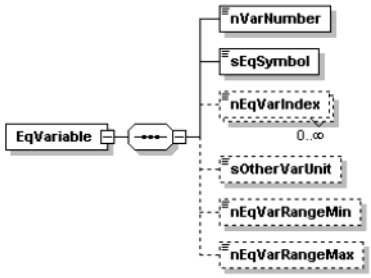

Figure 14. Structure of the **EqVariable** element for equation representation. **EqVariable** is a subelement of **Equation** [complex].

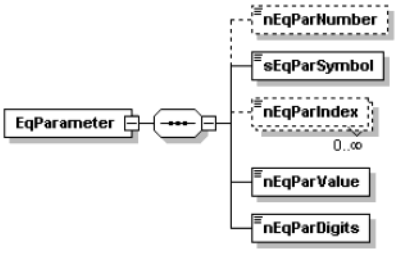

Figure 15. Structure of the **EqParameter** element for equation representation. **EqParameter** is a subelement of **Equation** [complex].

The structure of the subelement **EqVariable** [complex] within **Equation** [complex] is shown in Figure 14. The six subelements of **EqVariable** [complex]: **nVarNumber** [numerical, integer], **sEqSymbol** [string], **nEqVarIndex** [numerical, integer], **sOtherVarUnit** [string], **nEqVarRangeMin** [numerical, floating], and **nEqVarRangeMax** [numerical, floating], are analogous to those of **EqProperty** [complex] described above.

The structure of the subelement **EqParameter** [complex] within **Equation** [complex] is shown in Figure 15. **nEqParNumber** [numerical, integer] associates the parameter to a particular row and column in the covariance matrix. Parameters without **nEqParNumber** [numerical, integer] values do not contribute to the covariance, e.g., T_c in the Wagner eq 1, if it is established independent of the

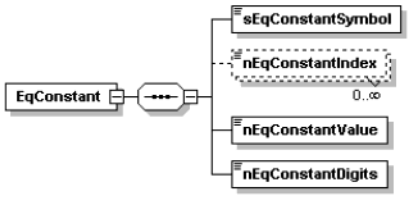

Figure 16. Structure of the **EqConstant** element for equation representation. **EqConstant** is a subelement of **Equation** [complex].

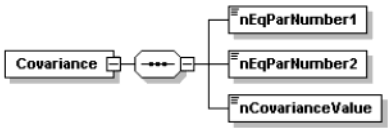

Figure 17. Structure of the **Covariance** element for equation representation.

represented vapor pressure values. **sEqParSymbol** [string] is used to map a symbol used in the equation definition (located on the Internet) to a particular parameter defined in the ThermoML file. **nEqParIndex** [numerical, integer] is used to map the parameter to a particular index in the equation definition. One index is used for a vector of parameters and two indexes for a matrix element (e.g., binary interaction parameters g represented as g_{ij} have indexes i and j). **nEqParValue** [numerical, floating] stores the numerical value of the parameter. **nEqParDigits** [numerical, integer] stores the total number of digits in **nEqParValue** [numerical, floating].

The structure of the subelement **EqConstant** [complex] within **Equation** [complex] is shown in Figure 16. **sEqConstantSymbol** [string] is used to map a symbol used in the equation definition to a particular constant defined in the ThermoML file. **nEqConstantIndex** [numerical, integer] is used to map the constant to a particular index in the equation definition. **nEqConstantValue** [numerical, floating] stores the numerical value of the constant. **nEqConstantDigits** [numerical, integer] stores the total number of digits in **nEqConstantValue** [numerical, floating].

The structure of the subelement **Covariance** [complex] within **Equation** [complex] is shown in Figure 17. This subelement is used to store the elements of the covariance matrix. **nEqParNumber1** [numerical, integer] is used to specify a particular equation parameter, **nEqParNumber** [numerical, integer] specified within **EqParameter** [complex]. **nEqParNumber2** [numerical, integer] is used to specify a second parameter defined within **EqParameter** [complex]. **nCovarianceValue** [numerical, floating] stores the covariance value for the two defined parameters.

The ThermoMLEquation Schema: Equation Definitions

The general structure of the schema (ThermoMLEquation) for equation definitions is shown in Figure 18. This template can be used by anyone who wishes to create an equation representation for use with ThermoML. The subelements of the generalized equation definition are defined in the following paragraphs. Indexes are not included explicitly in the ThermoMLEquation schema, but they are used for correct association of vector or matrix elements with numerical values in the ThermoML file, as demonstrated in the Supporting Information for this article.

The element **Version** [complex] is mandatory and provides for storage of the ThermoMLEquation version designation. The subelements of **Version** [complex] are **nVersionMajor** [numerical, integer] and **nVersionMinor** [numerical, integer]. For example, if the version number of ThermoMLEquation were 1.2, the "major" element would store the value "1" and the "minor" element would store the value "2".

sEqName [string] stores the name of the equation, such as *ThermoML.Wagner25Linear.VaporPressure*. (This is in accord with the naming convention suggested earlier.) In the case of the NIST/TRC library of equation representations, the names correspond to those provided in the element **eEqName** [enumeration] shown in Figure 11. The current enumeration list is provided in the text of the ThermoML schema included as Supporting Information. **sEqAltName** [string] stores alternative names for the same equation. **sEqDescription** [string] stores a general description of the equation in text format.

The element **EqReference** [complex] provides the means to store locations of information related to the equation description. **EqReference** [complex] has the same substructure as **Citation** [complex] shown in Figure 2 of the first paper in this series[1] and supports a wide variety of citations including those to journals, books, and Web sites.

Within any particular ThermoMLEquation representation, *properties*, *constraints*, and *variables*, as defined in ThermoML,[1] are not distinguished. In the context of mathematics, all of these quantities are "variables" and are so named in a ThermoMLEquation file as EqVariable [complex] as shown in Figure 18. Association of a particular *property*, *constraint*, or *variable* in a ThermoML data file with a "variable" in a ThermoMLEquation file is made through the assigned symbol and indexes (if needed) specified in **EqProperty** [complex] (Figure 12), **EqConstraint** [complex] (Figure 13), or **EqVariable** [complex] (Figure 14). This is why the element tag **sEqSymbol** [string] is the same for each of these complex elements.

The subelements of **EqVariable** [complex] in the ThermoMLEquation schema all provide additional information related to the particular variable. **sEqSymbol** [string] is the symbol used in the representation of the equation formulated with MathML. The element **sEqVarComment** [string] stores a text description of the particular variable. For example, **sEqSymbol** [string] might contain "T," while **sEqVarComment** [string] contains "temperature." This is a simple way to provide some description of symbols within the text of a ThermoMLEquation file. The element **IUPACSymbol** [complex] is provided to take advantage of the "presentation" elements of MathML and allows standard IUPAC symbols[13] to be associated with particular quantities in the equation.

The subelement of **IUPACSymbol** [complex], **mml:math**, indicates that this portion of the ThermoMLEquation schema allows importation of elements from MathML. In this case, the elements would be MathML presentation elements for representation of an IUPAC symbol.

The elements **EqParameter** [complex] and **EqConstant** [complex] within ThermoMLEquation have the same substructure as **EqVariable** [complex]. The subelements are analogous to those for **EqVariable** [complex]. A parameter in a ThermoML data file is associated with a particular equation parameter through use of the **sEqParSymbol** [string] element shown in Figure 15. Similarly, constants are associated with equation constants through the **sEqConstantSymbol** [string] (Figure 16).

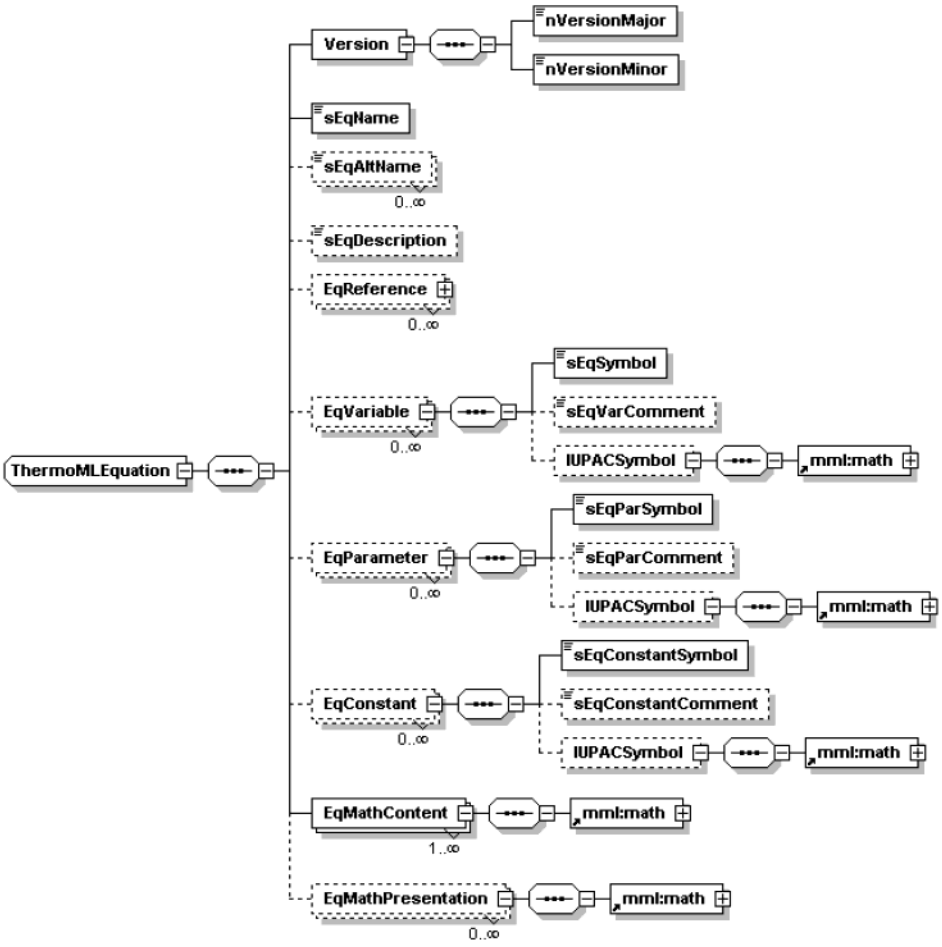

Figure 18. Structure of an equation definition.

The representation of the equation in MathML format is stored in the complex elements **EqMathContent** [complex] and **EqMathPresentation** [complex]. **EqMathContent** [complex] is used to store the mathematical meaning of the equation, and **EqMathPresentation** [complex] is used to store information for display. Examples of the representation of several equations with MathML are included in the Supporting Information.

Figure 19 shows a schematic representation of linking of *property*, *constraint*, *variable*, *parameter*, and *constant* information both within a ThermoML file and between a ThermoML and a ThermoMLEquation file. Particular schema elements through which the linking is accomplished are indicated on the figure.

Use Cases and ThermoML Schema Text

Examples of data files created with the ThermoML formats for predicted and critically evaluated data are included as Supporting Information. The examples are based upon studies published in the peer-reviewed literature involving predicted data,[14] equation representation,[15] and critically evaluated data together with equation representation including covariance information. The third example with covariance information was created at TRC.

The implementation of several types of fitting equations in ThermoMLEquation files is demonstrated in the Supporting Information. ThermoMLEquation files are provided for eleven equations ranging from a simple polynomial expansion to several relatively complex expressions for the residual Helmholtz energy as a function of temperature and density, such as[16]

$$1000A^{r}/(RT) = \sum_{i=1}^{nTerms} n_i \delta^{d_i} \tau^{t_i} \exp(e_i \delta^{p_i}) \qquad (3)$$

where A^{r} is the Helmholtz energy (in kJ·mol^{-1}) expressed as the difference between the real fluid and the ideal gas at the same temperature T and density ρ, δ is the reduced density (ρ divided by the critical density ρ_c), τ is the critical temperature T_c divided by T, and R is the gas constant. The constants e_i do not appear in the original formulation and were added here to provide a uniform presentation for all terms in the equation. The complete list of eleven equations and their full descriptions are provided in the Supporting Information.

The Role of ThermoML in Global Data Submission and Dissemination

The role of ThermoML in global submission and dissemination of experimental thermodynamic property data was described.[1] Guided data capture (GDC) software[17] was developed at TRC for mass-scale abstraction from the literature of experimental thermophysical and thermochemical property data and is available for free download from the Web.[18] Extension of GDC for capture of predicted and critically evaluated data is planned.

Following the peer-review process, authors are requested by the journal editors to download and use the GDC software to capture the experimental property data that have been accepted for publication. The output of the GDC

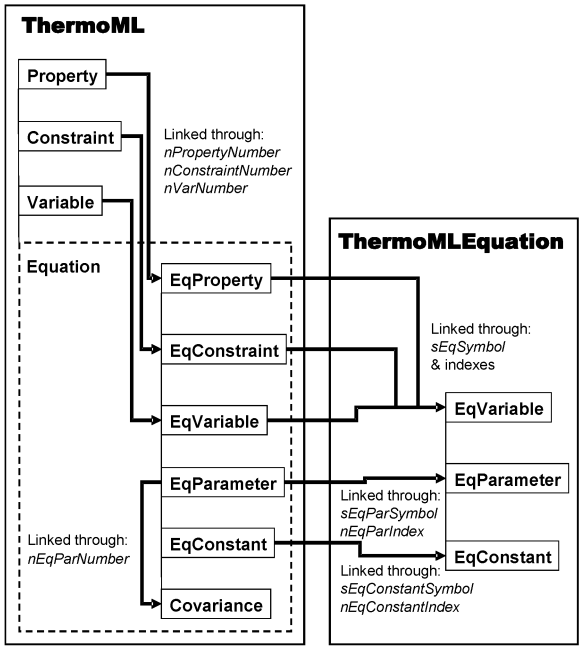

Figure 19. Linking of *property*, *constraint*, *variable*, *parameter*, and *constant* information both within a ThermoML file and between a ThermoML and ThermoMLEquation file. Particular schema elements through which the linking is accomplished are indicated on the figure. Within ThermoML, the elements **Property** [complex], **Constraint** [complex], **Variable** [complex], and **Equation** [complex] (shown in the figure) are immediate subelements of the major blocks *PureOrMixtureData* and *ReactionData*. The dotted box encloses the subelements of **Equation** [complex] involved in linking of equation information.

software is an electronic data file (plain-text file), which is submitted directly to TRC. The electronic data files are converted into ThermoML format with software (TransThermo) developed at TRC. Upon release of the manuscript for publication, the ThermoML files are posted on the public domain TRC Web site with unrestricted public access. This procedure has been established formally by the *Journal of Chemical and Engineering Data*.[19] Recently, this procedure was extended to the *Journal of Chemical Thermodynamics*.[20] Expansion of this operation to other journals in the field is under discussion.

TRC is currently working with a number of companies (producing software for chemical process design) in the development of software "readers" for ThermoML files. The purpose of these "readers" is transfer of thermodynamic data (ThermoML files) from the Web[21] directly to chemical process applications. Completion of these projects will represent creation of a new thermodynamic data delivery infrastructure allowing communication of the most recent data directly from the "producers" (experimentalists and data evaluators) to the users (process designers). These activities are conducted, in part, within the IUPAC project 2002-055-3-024 "XML-based IUPAC Standard for Experimental and Critically Evaluated Thermodynamic Property Data Storage and Capture."[22]

Future Extensions

This paper completes the development of the major elements of the ThermoML schema. However, the authors plan to extend ThermoML soon to include the IUPAC-NIST Chemical Identifier,[23] thermodynamic properties of polymers, and phase equilibria properties for electrolyte mixtures.

Acknowledgment

The authors express their appreciation to IUPAC for organizing meetings and facilitating discussions of the many issues covered in this publication within the project 2002-055-3-024. The authors also express their appreciation to Drs. S. E. Stein (NIST, Gaithersburg) and E. Königsberger (Murdoch University, Australia) for their valuable advice in the development of the extensions to the ThermoML format described in this paper. We thank the following people at NIST for their suggestions and comments related to this paper: Drs. T. C. Allison (Gaithersburg), R. A. Perkins (Boulder), and J. W. Magee (Boulder). We express our special appreciation to Dr. G. J. Rosasco (NIST, Gaithersburg) for his support and coordination of the project.

Supporting Information Available:

Examples are provided illustrating the use of ThermoML for representation of predicted and critically evaluated data as well as fitting equations. The complete current text of the ThermoML schema is included also as Supporting Information and is available on the Web (http://www.trc.nist.gov/ThermoML.xsd) or through direct request to the authors. This material is available free of charge via the Internet at http://pubs.acs.org.

Literature Cited

(1) Frenkel, M.; Chirico, R. D.; Diky, V. V.; Dong, Q.; Frenkel, S.; Franchois, P. R.; Embry, D. L.; Teague, T. L.; Marsh, K. N.; Wilhoit, R. C. ThermoML—An XML-Based Approach for Storage and Exchange of Experimental and Critically Evaluated Thermophysical and Thermochemical Property Data. 1. Experimental Data. *J. Chem. Eng. Data* **2003**, *48*, 2−13.

(2) Chirico, R. D.; Frenkel, M.; Diky, V. V.; Marsh, K. N.; Wilhoit, R. C. ThermoML—An XML-Based Approach for Storage and Exchange of Experimental and Critically Evaluated Thermophysical and Thermochemical Property Data. 2. Uncertainties. *J. Chem. Eng. Data* **2003**, *48*, 1344−1359.

(3) *Guide to the Expression of Uncertainty in Measurement* (International Organization for Standardization, Geneva, Switzerland, 1993). This *Guide* was prepared by ISO Technical Advisory Group 4 (TAG 4), Working Group 3 (WG 3). ISO/TAG 4 has as its sponsors the BIPM, IEC, IFCC (International Federation of Clinical Chemistry), ISO, IUPAC (International Union of Pure and Applied Chemistry), IUPAP (International Union of Pure and Applied Physics), and OIML. Although the individual members of WG 3 were nominated by the BIPM, IEC, ISO, or OIML, the *Guide* is published by ISO in the name of all seven organizations.

(4) *U. S. Guide to the Expression of Uncertainty in Measurement*, ANSI/NCSL Z540-2-1997, ISBN 1-58464-005-7; NCSL International: Boulder, CO, 1997.

(5) Taylor, B. N.; Kuyatt, C. E. *Guidelines for the Evaluation and Expression of Uncertainty in NIST Measurement Results*; NIST Technical Note 1297; NIST: Gaithersburg, MD, 1994.

(6) Wilhoit, R. C.; Marsh, K. N. Future Directions for Data Compilation. *Int. J. Thermophys.* **1999**, *20*, 247−255.

(7) Frenkel, M. Dynamic Compilation: A Key Concept for Future Thermophysical Data Evaluation. In *Forum 2000: Fluid Properties for New Technologies − Connecting Virtual Design with Physical Reality*; Rainwater, J. C., Friend, D. G., Hanley, H. J. M., Harvey, A. H., Holcomb, C. D., Laesecke, A., Magee, J. W., Muzny, C., Eds.; NIST Special Publication 975; NIST: Gaithersburg, MD, 2001; pp 83−84.

(8) http://www.trc.nist.gov/properties.htm.

(9) Sandhu, P. *The MathML Handbook*; Charles River Media, Inc.: Hingham, MA, 2003. See also: www.w3.org/Math/.

(10) See: http://www.w3.org/.

(11) *XML SPY v. 4.4 u*. ALTOVA GmbH and ALTOVA, Inc.: 1998−2002.

(12) Frenkel, M.; Hong, X.; Dong, Q.; Yan, X.; Chirico, R. D. *Densities of Halohydrocarbons*. Landolt-Börnstein Series; Springer-Verlag: Berlin, 2003; Vol. IV/8J.

(13) Mills, I.; Cvitas, T.; Homann, K.; Kallay, N. and Kuchitsu, K. *Quantities, Units and Symbols in Physical Chemistry (The Green Book)*; Blackwell Science: Oxford, 1993.

(14) Steele, W. V.; Chirico, R. D.; Cowell, A. B.; Nguyen, A.; Knipmeyer, S. E. Possible Precursors and Products of Deep Hydrodesulfurization of Gasoline and Distillate Fuels. Part 2. The Thermodynamic Properties of 2,3-Dihydrobenzo[*b*]thiophene. *J. Chem. Thermodyn.* **2003**, *35*, 1253−1276.

(15) Chirico, R. D.; Knipmeyer, S. E.; Nguyen, A.; Steele, W. V. Thermodynamic Properties of the Methylpyridines. Part 2. Vapor Pressures, Heat Capacities, Critical Properties, Derived Thermodynamic Functions between the Temperatures 250 K and 560 K, and Equilibrium Isomer Distributions for All Temperatures > 250 K. *J. Chem. Thermodyn.* **1999**, *31*, 339−378.
(16) Lemmon, E. W., Jacobsen, R. T. An International Standard Formulation for the Thermodynamic Properties of 1,1,1-Trifluoroethane (HFC-143a) for Temperatures from 161 to 450 K and Pressures to 50 MPa. *J. Phys. Chem. Ref. Data* **2000**, *29*, 521−552.
(17) Diky, V. V.; Chirico, R. D.; Wilhoit, R. C.; Dong, Q.; Frenkel, M. Windows-Based Guided Data Capture Software for Mass-Scale Thermophysical and Thermochemical Property Data Collection. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 15−24.
(18) http://www.trc.nist.gov/GDC.html.
(19) Marsh, K. N. New Process for Data Submission and Dissemination. *J. Chem. Eng. Data* **2003**, *48*, 1.
(20) Electronic Data Submission to the NIST Thermodynamics Research Center. *J. Chem. Thermodyn.* **2004**, *36*, iv.
(21) http://www.trc.nist.gov/ThermoML.html.
(22) http://www.iupac.org/projects/2002/2002-055-3-024.html.
(23) http://www.iupac.org/projects/2000/2000-025-1-800.html.

Received for review March 18, 2004. Accepted March 22, 2004. The authors express their appreciation to IUPAC for providing travel funds.

JE049890E