The Formal Design Models of a Universal Array (UA) and its Implementation

Yingxu Wang, University of Calgary, Canada
Jason Huang, University of Calgary, Canada
Jingsheng Lei, Shanghai University of Electrical Power, China

ABSTRACT

Arrays are one of the most fundamental and widely applied data structures, which are useful for modeling both logical designs and physical implementations of multi-dimensional data objects sharing the same type of homogeneous elements. However, there is a lack of a formal model of the universal array based on it any array instance can be derived. This paper studies the fundamental properties of Universal Array (UA) and presents a comprehensive design pattern. A denotational mathematics, Real-Time Process Algebra (RTPA), allows both architectural and behavioral models of UA to be rigorously designed and refined in a top-down approach. The conceptual model of UA is rigorously described by tuple- and matrix-based mathematical models. The architectural models of UA are created using RTPA architectural modeling methodologies known as the Unified Data Models (UDMs). The physical model of UA is implemented using linear list that is indexed by an offset pointer of elements. The behavioral models of UA are specified and refined by a set of Unified Process Models (UPMs). As a case study, the formal UA models are implemented in Java. This work has been applied in a number of real-time and nonreal-time systems such as compilers, a file management system, the real-time operating system (RTOS+), and the ADT library for an RTPA-based automatic code generation tool.

Keywords: Abstract Data Types, Design Frameworks, Formal Design Models, Software Engineering, Software Science, System Architecture Specification, System Behavior Specification, Unified Data Models, Universal Arrays

1. INTRODUCTION

Although it is difficult to image higher dimensional structures in data and architectural modeling when the number of dimensions is greater than three, there are fundamental needs in science and engineering to model a wide range of natural entities and abstract artifacts in higher dimensions. For example, in physics, despite the natural world is conveniently perceived as three dimensional, the abstract world is modeled by 11 dimensions (11-D) or even higher ones.
according to the string theories (Becker, 2007). In cognitive informatics, the human behavioral space is a 4-dimensional structure known as object, action, space, and time (Wang, 2007). In computational linguistics, a sentence, its syntactic structure, and semantic space are recognized by 1-D, 2-D, and 5-D models, respectively (Wang, 2009c). In computer science, there are numerals entities and abstract data models that are multi-dimensional arrays or matrices such as in numerical methods, computational intelligence, and weather predication (Chapra & Ganale, 2002; Stubbs & Webre, 1985).

An array is a linear data structure with homogeneous elements accessible by random or sequential addressing. The mathematical model of arrays is a vector (1-D array) or a tuple of matrices (n-D arrays) (Stubbs & Webre, 1985; Matlab, 2010). Arrays are the most fundamental and widely used data structures to model both logical designs and physical implementations of multi-dimensional data objects that share the same type of elements. Arrays are the most fundamental and widely used data structures in computing and everyday life. Various important data objects can be modeled and implemented by arrays such as a set of numbers, a list of names, a relational database, and a dictionary of words in a language (McDermid, 1991; Wiener & Pinson, 2000). Arrays play a central role in Matlab where the fundamental data models are vectors and matrices that can be implemented by 1-D and n-D arrays in computing, respectively (Matlab, 2010).

Data object modeling is a process to creatively extract and abstractly represent a real-world problem by data models based on the constraints of given computing resources (Wang, 2008b). An arbitrary array can be formally modeled by an Abstract Data Type (ADT) (Guttag, 1977; Broy et al., 1984; Cardelli & Wegner, 1985; Stubbs & Webre, 1985), which is a logical model of a complex and/or user defined data structure with a set of predefined operations. Using types to model real-world entities can be traced back to the mathematical thought of Bertrand Russell in 1900s (Russell, 1903) and George Cantor in 1932 (Lipschutz & Lipson, 1997). A number of ADTs have been identified in computing and system modeling such as stack, queue, sequence, record, array, list, tree, file, and graph (Broy et al., 1984; Mitchell, 1990; McDermid, 1991; Wang, 2007; Wang, Ngolah, Tan, Tian, & Sheu, 2010). ADTs possess the following properties: (i) An extension of type constructions by integrating both data structures and functional behaviors; (ii) A hybrid data object modeling technique that encapsulates both user defined data structures and allowable operations on them; (iii) The interface and implementation of an ADT are separated where detailed implementation of the ADT is hidden to applications that invoke the ADT and its predefined operations.

In order to formally model the universal array as an ADT, an expressive denotational mathematics, Real-Time Process Algebra (RTPA) (Wang, 2002, 2007, 2008a, 2008b, 2008c, 2008d, 2009a; Wang, Tan, & Ngolah, 2010), is adopted, which allows both architectural and behavioral models of universal array to be rigorously designed and implemented in a top-down approach. According to the RTPA methodology for system modeling and refinement, a universal array can be rigorously modeled using two fundamental techniques known as the unified data models and the unified process models (Wang, 2007).

Definition 1. A Unified Data Model (UDM) is a generic architectural model for a software system as well as its internal control structures and its interfaces with hardware components, which can be rigorously modeled and refined as an n-tuple, i.e.:

\[
UDM \triangleq \prod_{i=1}^{n} (s_i \mid \forall e \in S_i, p_i(e))
\]
LZW Encoding in Genetic Algorithm
www.igi-global.com/chapter/lzw-encoding-genetic-algorithm/74933?camid=4v1a

A Theory of Program Comprehension: Joining Vision Science and Program Comprehension
www.igi-global.com/article/theory-program-comprehension/2793?camid=4v1a