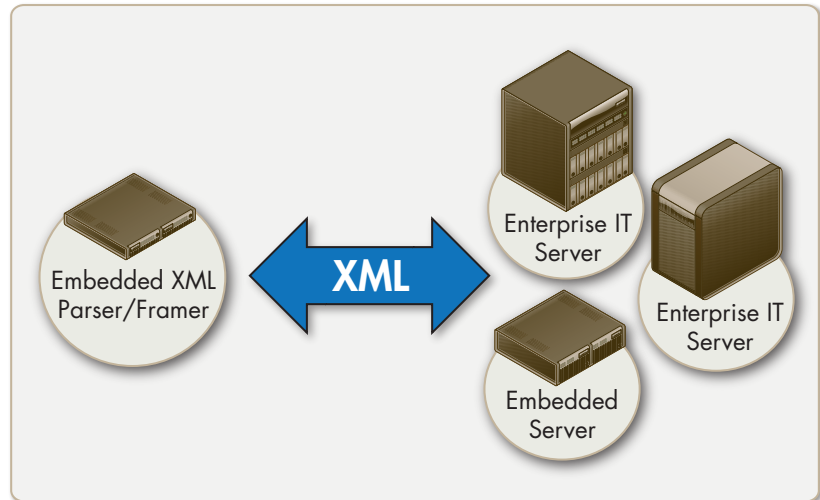


RomXML®

Embedded XML Parsing and Framing Toolkit

RomXML Benefits

- Decrease time to market by leveraging proven embedded XML technology
- Lightweight Parsing and Framing technology
- Translate XML syntax to and from embedded internal C-language structures
- Ability to create and read XML based documents without carrying the overhead of general purpose XML tools
- Utilizes RxSchema to define both C-language internal storage format and the set of XML elements and attributes for XML-based data interchange
- RomXML Tagbuilder utility program drastically reduces coding effort by analyzing object definitions and produces C-language source with structures pre-defined and ready to use with Parsing and Framing functions
- Small RAM/ROM footprint
- ANSI-C Implementation
- Compilation switches for size and speed trade-offs
- Ability to use standalone or in conjunction with Allegro's RomPager suite of products
- Highly portable



Allegro's RomXML Parsing and Framing Toolkit drastically decreases the time and effort needed to implement eXtensible Markup Language (XML) in your embedded application. XML offers a processor-independent method to encode data for interchange between diverse systems and is based on a set of rules for the construction of tag-delimited information.

RomXML

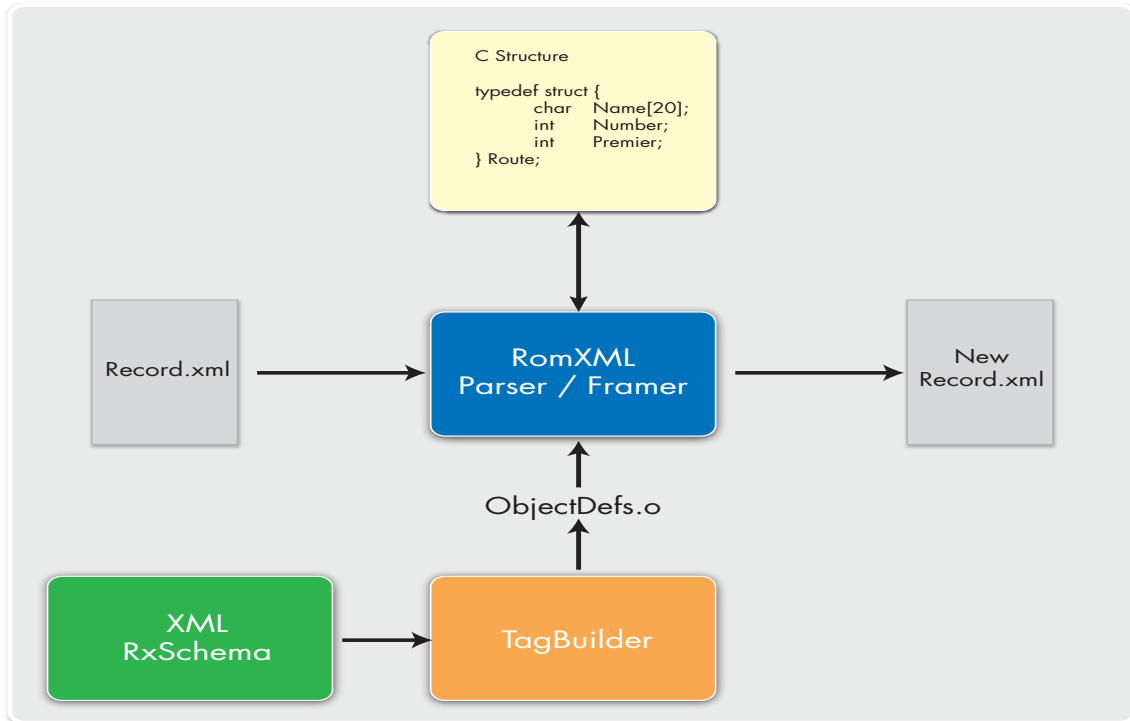
When designing embedded systems, it is useful to leverage standards based technology to exchange data between your embedded device and general purpose computing desktops or workstations. Whether you are engaged in developing a unique man-machine interface or a sophisticated machine-machine interface, a standard interchange format such as XML simplifies and reduces your development efforts. However, implementing a general purpose XML parser will likely prove to be impractical. For typical embedded applications, the tag set or schema does not change. The primary need for an embedded XML Parser or Framer is to efficiently translate data between XML syntax and internal storage (typically a C structure). The Allegro RomXML Parsing and Framing toolkit provides a lightweight translation between pre-defined C-language structures and XML-based representations.

XML Parsing and Framing

Within enterprise environments, there are two common methods employed for dealing with XML. One approach is a method called Simple API for XML (SAX). In the SAX approach, as a document is parsed, XML elements and attributes are passed to a user program for interpretation. The XML information and underlying structure are discarded in this parsing method once the data is passed to the user program. While this method doesn't use much memory, it requires the user program to deal with XML in a sequential manner rather than as a whole entity. Since the XML syntax and underlying structure are discarded, SAX style parsers do not provide much support for the framing process.

Allegro Software Development Corporation
1740 Massachusetts Avenue
Boxborough, Massachusetts
01719

+1 (978) 264-6600
www.allegrosoft.com



The Second approach typically utilized in an enterprise environment is a method called the Document Object Model (DOM). A DOM parser recognizes that an XML document is a hierarchical data format and stores a tree structure representing both the underlying XML information and data. An API is provided to traverse the tree structure in any order and allows access to the entire XML document. Framing a new document is very straight forward since the XML data and overall structure are retained. However, the DOM method for parsing and framing documents is very memory intensive often utilizing over twice the memory of the original document.

The RomXML toolkit is designed for embedded devices that have limited resources and wish to use a more powerful method than SAX, without the overhead of DOM. For embedded applications, devices do not need the ability to parse and frame general purpose XML documents. However, embedded devices do require the ability to efficiently translate device specific data between an internal format (such as a C structure) and XML syntax. This means the schema (XML element definitions) for a given document do not change and can be specified at compile time. The RomXML toolkit provides a light weight translation between pre-defined C-Language structures and XML formatted representations. This enables an embedded device to exchange data as well-formed XML documents without the overhead of general-purpose XML translators.

TagBuilder

As part of the RomXML toolkit, the TagBuilder compiler further decreases your development effort when utilizing XML in your embedded design. RomXML uses a special set of tags (RxSchema) to define an XML object. The RxSchema language allows an XML object to be defined with both C-language internal storage format and the set of XML elements and attributes used with XML-based data exchange. The RomXML TagBuilder compiler analyzes the RomXML RxSchema object definitions and produces an object definition file in C-language source that is compiled with the RomXML Parser and Framer code. The definition file contains the transfer tables that the runtime RomXML code utilizes to perform specific translations.

Flexible Design

The RomXML runtime is delivered in ANSI-C, is highly portable and offers a rich API for your development team to handle XML objects. RomXML can handle XML datastreams transmitted with your proprietary communications methodologies or in connection with Allegro's RomPager suite of products. When used together with the rich RomPager suite, web services such as SOAP and XML-RPC are also available for use in your design.