WIKIPEDIA

Seed7

Seed7 is an extensible general-purpose programming language designed by Thomas Mertes. It is syntactically similar to Pascal and Ada. Along with many other features, it provides an extension mechanism.[2] Seed7 supports introducing new syntax elements and their semantics into the language, and allows new language constructs to be defined and written in Seed7.[3] For example, programmers can introduce syntax and semantics of new statements and user defined operator symbols. The implementation of Seed7 differs significantly from that of languages with hard-coded syntax and semantics.

Seed7

Paradigm	multi-paradigm: extensible, object-oriented, imperative, structured, generic, reflective
Designed by	Thomas Mertes
First appeared	2005
Stable release	2020-12-08 / 31 days ago[1]
Typing discipline	static, strong, safe, nominative, manifest
OS	Cross-platform: BSD, Linux, OS X, Unix, Windows
License	GPL, LGPL (for the runtime library)
Filename extensions	.sd7, .s7i
Website	seed7.sourceforge.net (http://seed7.sourceforge.net)
Major implementations	
open source reference implementation	
Influenced by	
Pascal, Modula-2, Ada, ALGOL 68, C, C++, Java	

Contents

Features

Seed7 supports the programming paradigms: imperative, object-oriented (OO), and generic. It also supports features such as call by name, multiple dispatch, function overloading, operator overloading, exception handling and arbitrary-precision arithmetic.

Major features include:

- User defined statements and operators
- Abstract data types
- Templates without special syntax
- Object-oriented with interfaces and multiple dispatch
- Static typing
- May be interpreted or compiled
- Source code portability
- Runs under BSD, Linux, Mac OS X, Unix, Windows

Several programming language concepts are generalized:

- Type declarations (which assign a name to a type) and function definitions take the form of constant definitions.
- Compile time expressions can execute user-defined functions.
- Overloading and object-orientation (with multiple dispatch) are seen as common concepts. They just happen at different times: compile time and run time, respectively.
- Type names and type descriptions can be used as parameter and function result.
- Functions, which are executed at compile time, can be used to define objects.
- Templates are written as compile time functions with type parameters.
- Arrays, hash maps and structs are not a hard-coded feature. Instead they are defined as abstract data type in libraries.
- Parser and interpreter are part of the runtime library.
- UTF-32 Unicode support. This avoids problems of variable-length encodings like UTF-8 and UTF-16.

The Seed7 project includes both an interpreter and a compiler. The interpreter starts programs very quickly, supporting fast program development. The compiler uses the parser and reflection interfaces from the run-time library to generate a C program, which is subsequently compiled to machine code. Compiled Seed7 programs can have comparable performance to C programs.[4]

Libraries

Seed7 has many libraries, covering areas including containers, numeric functions, lexical analysis, file manipulation, networking (sockets, Transport Layer Security (TLS/SSL), Hypertext Transfer Protocol (HTTP), HTTP Secure (HTTPS), File Transfer Protocol (FTP), Simple Mail Transfer Protocol (SMTP), etc.), graphics, pixmap and vector fonts, database independent API, Common Gateway Interface (CGI) support, data compression, archive files (tar, zip, cpio, ar, rpm), character encoding, time and date handling, XML processing, message digests and more.[5] These libraries reduce the need to use unportable operating system features and third-party libraries (which might not always be present) directly. Seed7 libraries contain abstraction layers for hardware, operating system and third-party libraries, e.g. graphic and database libraries. In other words, no changes are needed to move Seed7 programs between different processors or operating systems.

TLS library

Seed7 has its own implementation of Transport Layer Security.[6] The library includes AES and elliptic-curve cryptography.

Database abstraction API

Seed7 provides a library with a database independent API.[7] Programs can connect to MySQL, MariaDB, SQLite, PostgreSQL, Oracle, Open Database Connectivity (ODBC), Firebird, InterBase, Db2 and SQL Server databases. Independent from the database prepared statements can be created, bind variables can be used, the statements can be executed and result sets can be fetched.

History

Seed7 is based on MASTER, an extensible programming language described in the diploma and doctoral theses of Thomas Mertes.[8][9] Most of the original ideas of MASTER, such as user defined statements and operators, can be found in Seed7. A precompiler, to translate MASTER to Pascal, was proposed, but unimplemented, in the original project. In 1989, development began on an interpreter for MASTER, named HAL. In 2005, the MASTER and HAL projects were released as open source under the Seed7 project name. Since then new versions have been released every two or three weeks. As of version 2013-09-08 the Seed7 project contains more than 300,000 source lines of code and several hundred pages of documentation.

Extension mechanism

An extension includes two parts: a syntax definition, giving a template for the new syntactic form, and a standard Seed7 function, used to define the semantics.[2]

Syntax definition

The syntax definition uses the Seed7 Structured Syntax Description (S7SSD). A S7SSD statement like

```
$ syntax expr: .(). + .()  is -> 7;
```

specifies the syntax of the `+` operator. The right arrow `->` describes the associativity: Binding of operands from left to right. With `7` the priority of the `+` operator is defined. The syntax pattern `.(). + .()` is introduced and delimited with dots (`.`). Without dots the pattern is `() + ()`. The symbol `()` is a nonterminal symbol and `+` is a terminal symbol.[10] The S7SSD does not distinguish between different nonterminal symbols. Instead it only knows one nonterminal symbol: `()`.

Semantic extension

The definition of the `+` operator for complex numbers is just a function definition:

```
const func complex: (in complex: summand1) + (in complex: summand2) is func
  result
    var complex: sum is complex.value;
  begin
    sum.re := summand1.re + summand2.re;
    sum.im := summand1.im + summand2.im;
  end func;
```

References

1. Mertes, Thomas (8 December 2020). "New Seed7 Release 2020-12-08" (https://sourceforge.net/p/seed7/news/2020/12/new-seed7-release-2020-12-08/). Retrieved 8 December 2020 – via SourceForge.
2. Daniel Zingaro, "Modern Extensible Languages" (http://www.cas.mcmaster.ca/sqrl/papers/SQRLreport47.pdf), SQRL Report 47 McMaster University (October 2007), page 16 (alternate link (http://www.danielzingaro.com/extensible.pdf)).
3. Abrial, Jean-Raymond and Glässer, Uwe, "Rigorous Methods for Software Construction and Analysis" (https://books.google.com/books?id=hc8FIwfSQi8C&pg=PA166#v=onepage&q&f=fal

se), ISBN 978-3-642-11446-5, Springer, 2010, page 166.

4. Stadfeld, Paul (1 April 2010). "The Quest for the Ultimate Cycle (includes a performance comparison between Python, Seed7 and C)" (https://web.archive.org/web/20131025193458/http://mensanator.com/mensanator/cycle/ultimate_cycle.htm). Archived from the original (http://mensanator.com/mensanator/cycle/ultimate_cycle.htm) on 25 October 2013. Retrieved 19 August 2019.
5. Seed7 libraries (http://seed7.sourceforge.net/libraries)
6. A Transport Layer Security (TLS) library written in Seed7 (http://seed7.sourceforge.net/libraries/tls.htm)
7. Database abstraction API (http://seed7.sourceforge.net/manual/database.htm)
8. Mertes, Thomas, "Entwurf einer erweiterbaren höheren Programmiersprache", Diploma thesis Vienna University of Technology (1984).(Abstract (https://web.archive.org/web/20120205234431/https://www.complang.tuwien.ac.at/library/ibv/detail.php?nr=1082))
9. Mertes, Thomas, "Definition einer erweiterbaren höheren Programmiersprache" (http://seed7.sourceforge.net/thesis/doctorate), Doctoral thesis Vienna University of Technology (1986). (Abstract (https://web.archive.org/web/20130509083209/http://www.complang.tuwien.ac.at/library/ibv/detail.php?nr=1083))
10. David Gudeman (March 26, 2015), "The Seed7 Programming Language" (In Seed7 you can specify a syntax like this) (http://unobtainabol.blogspot.co.at/2015/03/the-seed7-programming-language.html)

External links

- Official website (http://seed7.sourceforge.net) - Homepage with FAQ, manual, screenshots, examples, library descriptions, benchmarks and a set of algorithms
- Seed7 at GitHub (https://github.com/ThomasMertes/seed7)
- Download Seed7 (https://sourceforge.net/projects/seed7/files) from its main repository at Sourceforge
- Seed7 at Rosetta Code (http://rosettacode.org/wiki/Category:Seed7) - Contains many Seed7 examples
- The Quest for the Ultimate Cycle (https://web.archive.org/web/20131025193458/http://mensanator.com/mensanator/cycle/ultimate_cycle.htm) explores the 3n+C extension of the Collatz Conjecture with Seed7 programs
- Blog by Remo Laubacher (http://www.mesch.ch/team/remo-laubacher/#panel-4): Statically linked Linux executables with GCJ, Seed7 and haXe (http://www.codeblog.ch/2011/06/statically-linked-linux-executables) (2011)
- Blog by David Gudeman (https://plus.google.com/109271233617507214410/posts) The Seed7 Programming Language (http://unobtainabol.blogspot.co.at/2015/03/the-seed7-programming-language.html) (2015)
- A FreeBSD port (http://www.freshports.org/lang/seed7) / see also here (http://www.freebsd.org/cgi/cvsweb.cgi/ports/lang/seed7), maintained by Pietro Cerutti (http://ch.linkedin.com/in/pietrocerutti)
- An OpenBSD port (http://openports.se/lang/seed7), provided by Brian Callahan
- A Seed7 package for openSUSE/Fedora (https://build.opensuse.org/package/show/home%3Azhonghuaren/seed7), (see also here (https://software.opensuse.org/package/seed7) or here (https://software.opensuse.org/download.html?project=home%3Azhonghuaren&package=seed7))
- A discussion where Seed7 is described as language where new syntax can actually be defined by language users (http://www.linuxtoday.com/news_story.php3?ltsn=2009-10-21-023-35-OS-HE&tbovrmode=3)

- Seed7 at "Fossies" - the Fresh Open Source Software Archive (https://fossies.org/linux/misc/seed7_05_20200929.tgz/)

Retrieved from "https://en.wikipedia.org/w/index.php?title=Seed7&oldid=993300043"