



Hewlett Packard  
Enterprise

# CHAPEL RELEASE NOTES, 1.25.1 / 1.26.0: DOCUMENTATION UPDATES



Chapel Team

December 9, 2021 / March 31, 2022



# DOCUMENTATION UPDATES

## Background and This Effort

---

### Background:

- Chapel's documentation has been hosted online for many releases now
  - <https://chapel-lang.org/docs/>
- However, by nature, documentation can always benefit from improvements or reorganizations

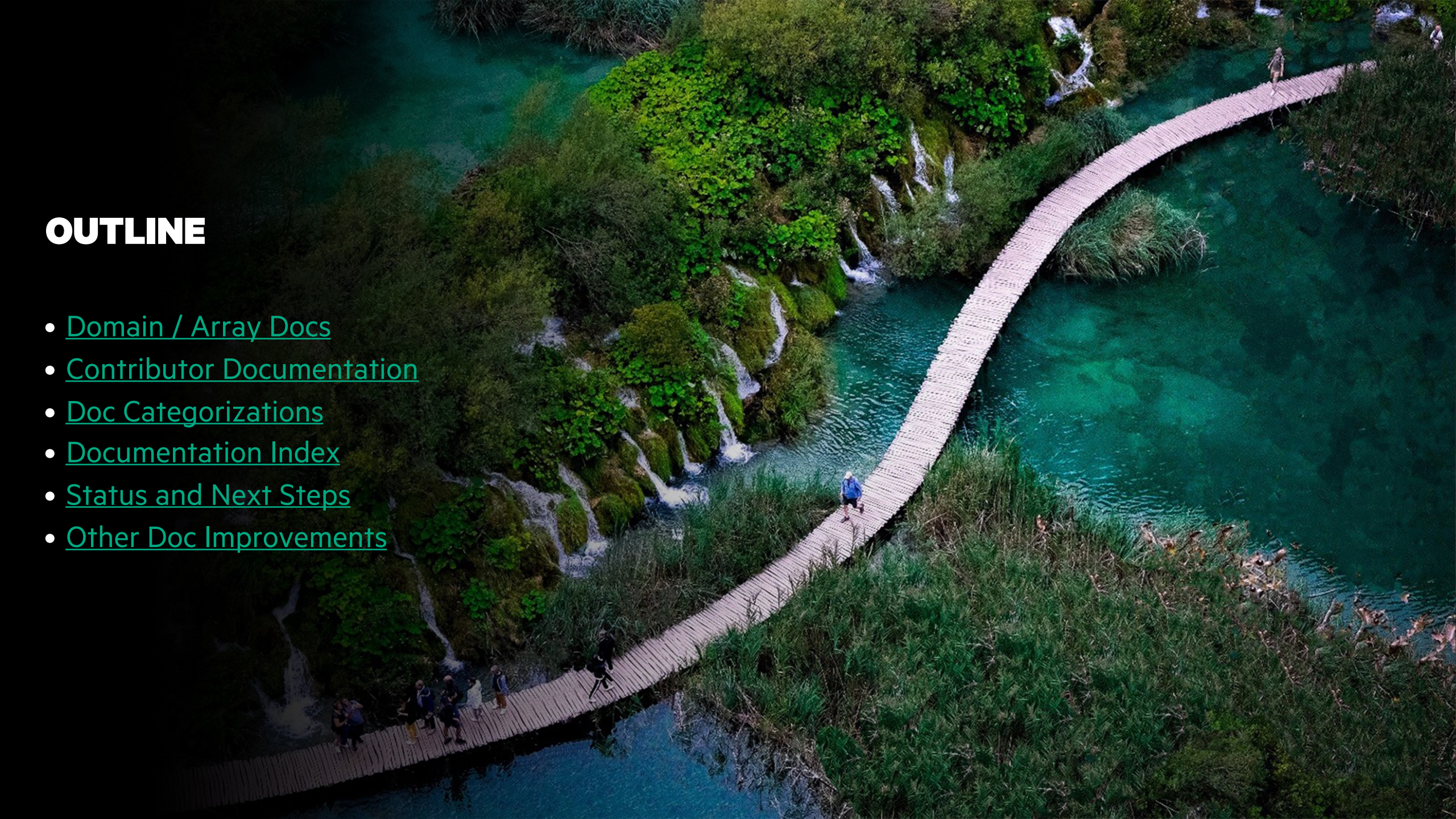
### This Effort:

- Made several improvements to content and organization
  - merged domain and array “built-in types and functions” sections into the language specification
  - moved contributor documentation online
  - categorized the language specification chapters and package modules
  - added the index to the sidebar



# OUTLINE

- [Domain / Array Docs](#)
- [Contributor Documentation](#)
- [Doc Categorizations](#)
- [Documentation Index](#)
- [Status and Next Steps](#)
- [Other Doc Improvements](#)



# DOCUMENTATION UPDATES

## Domain and Array Documentation

**Background:** The language specification has overlapping content with “built-in types and functions”

- the language specification is written and maintained manually
- “built-in types and functions” is generated by ‘chpdoc’ from code, so is generally more accurate

**This Effort:** Moved ‘chpdoc’-generated domain/array docs into the spec, replacing redundant content

The image displays two side-by-side screenshots of the Chapel documentation website. The left screenshot shows the 'Built-in Types and Functions' page. The sidebar menu on the left includes 'Hello World Variants', 'Primers', 'Language Specification', and 'Built-in Types and Functions'. Under 'Built-in Types and Functions', there is a list of items: 'owned', 'Bytes', 'Ranges', 'shared', '~~Domain and Array Operations~~', 'Synchronization Variables', 'Strings', and 'Tuples'. The main content area is titled 'Built-in Types and Functions' and contains the text: 'The following sections describe built-in language features which are amenable to being documented using chpdoc:'. Below this text is a bulleted list: '• owned', '• Bytes', '• Ranges', '• shared', '• ~~Domain and Array Operations~~', '• Synchronization Variables', '• Strings', and '• Tuples'. The right screenshot shows the 'type array' page. The sidebar menu on the left includes 'WRITING CHAPEL PROGRAMS', 'Quick Reference', 'Hello World Variants', 'Primers', 'Language Specification', 'Introductory Material', 'Language Basics', 'Code Structures', 'Composite Types', 'Tuples', 'Classes', 'Records', 'Unions', 'Ranges', and 'Domains'. The 'Language Specification' item is selected. The main content area is titled 'type array' and contains the text: 'The array type'. Below this text are several procedure definitions: 'proc eltType', 'proc idxType', 'proc intIdxType', 'proc rankparam', and 'proc indices'. Each procedure definition is followed by a brief description of its purpose.

**Impact:** Less redundancy, improved accuracy

**Next Steps:** Do the same for the remaining seven sections



# DOCUMENTATION UPDATES

## Contributor Documentation

**Background:** For years, we have had documentation targeting project contributors

- has only existed as text files within the GitHub repository

**This Effort:** Moved this contributor documentation online

**Impact:** Easier to find, read, search

**Next Steps:** Continue to improve content, organization

The screenshot shows the Chapel Documentation website. The header includes 'Chapel Documentation' and 'version 1.26'. A search bar is present. The main navigation menu lists 'COMPILING AND RUNNING CHAPEL' (Quickstart Instructions, Using Chapel, Platform-Specific Notes, Technical Notes, Tools) and 'Docs for Contributors' (Best Practices for Contributors, Compiler Library API Docs, Query Framework, Parsing, Resolution). The current page is 'Compiler Library API Docs', which includes a description, a 'Conceptual Guide' with a link to 'Query Framework', and 'Symbols by Namespace' with links to 'Parsing', 'Resolution', 'Types', 'Untyped AST (uAST)', and 'Base'.

The screenshot shows the 'Best Practices for Contributors' page. The header includes 'Chapel Documentation' and 'version 1.26'. The breadcrumb trail is 'Docs for Contributors » Best Practices for Contributors'. The page content includes a description of the directory, sections for 'Getting Started', 'Contributor Info and Getting started with Chapel and the Developer Certificate of Origin (DCO)', 'Compiler documentation', 'Compiler Library API Docs', 'Tips On Debugging The Compiler', and 'Examining/Debugging Compiler IR'. A code block shows the path: `$CHPL_HOME/doc/rst/developer/implementation/compilerOverview`.



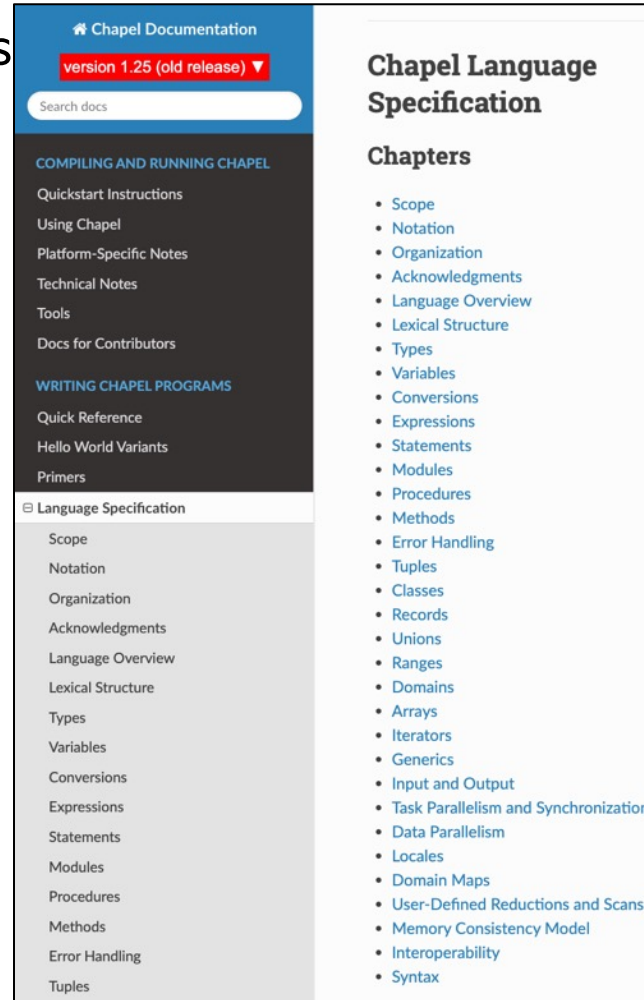
# DOCUMENTATION UPDATES

## Documentation Categorizations

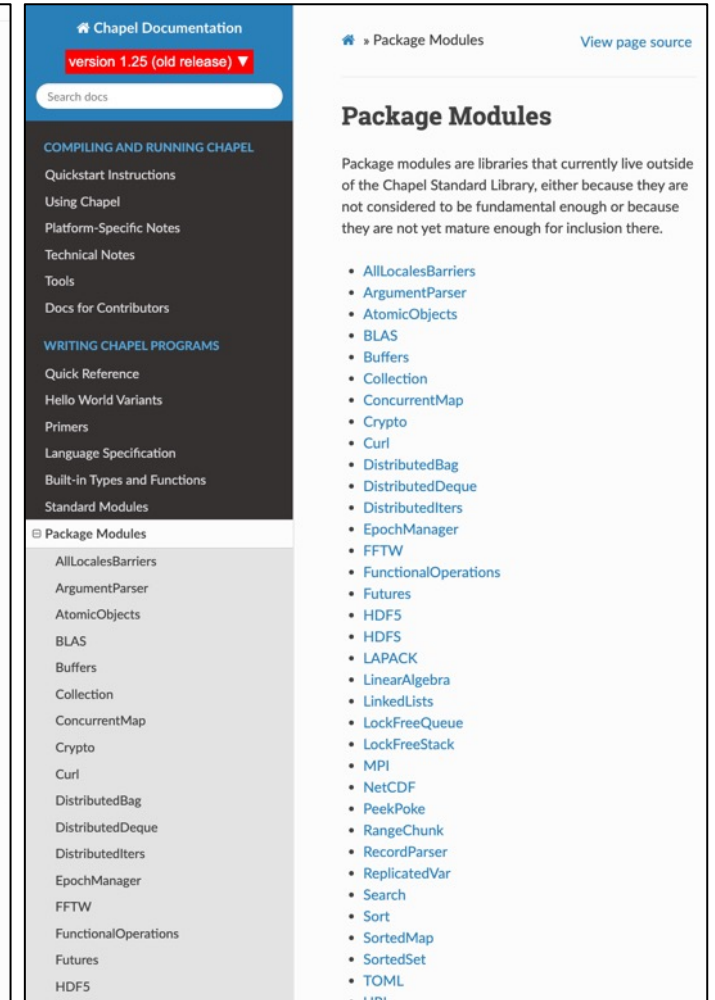
**Background:** Language specification chapters and package modules were simple lists

**This Effort:** Sorted each into categories

**Impact:** Table of contents and sidebar are more hierarchal, less of a laundry list



The screenshot shows the 'Chapel Language Specification' page. The header includes 'Chapel Documentation' and 'version 1.25 (old release)'. A search bar is present. The left sidebar lists categories: 'COMPILING AND RUNNING CHAPEL', 'WRITING CHAPEL PROGRAMS', and 'Language Specification'. The main content area is titled 'Chapel Language Specification' and 'Chapters', followed by a bulleted list of topics such as Scope, Notation, Organization, Acknowledgments, Language Overview, Lexical Structure, Types, Variables, Conversions, Expressions, Statements, Modules, Procedures, Methods, Error Handling, Tuples, Classes, Records, Unions, Ranges, Domains, Arrays, Iterators, Generics, Input and Output, Task Parallelism and Synchronization, Data Parallelism, Locales, Domain Maps, User-Defined Reductions and Scans, Memory Consistency Model, Interoperability, and Syntax.



The screenshot shows the 'Package Modules' page. The header includes 'Chapel Documentation' and 'version 1.25 (old release)'. A search bar is present. The left sidebar lists categories: 'COMPILING AND RUNNING CHAPEL', 'WRITING CHAPEL PROGRAMS', and 'Package Modules'. The main content area is titled 'Package Modules' and contains a paragraph explaining that package modules are libraries that currently live outside of the Chapel Standard Library. Below this is a bulleted list of module names: AllLocalesBarriers, ArgumentParser, AtomicObjects, BLAS, Buffers, Collection, ConcurrentMap, Crypto, Curl, DistributedBag, DistributedDeque, DistributedFilters, EpochManager, FFTW, FunctionalOperations, Futures, HDF5, LAPACK, LinearAlgebra, LinkedLists, LockFreeQueue, LockFreeStack, MPI, NetCDF, PeekPoke, RangeChunk, RecordParser, ReplicatedVar, Search, Sort, SortedMap, SortedSet, TOML, and URI.



# DOCUMENTATION UPDATES

## Documentation Categorizations

**Background:** Language specification chapters and package modules were simple lists

**This Effort:** Sorted each into categories

**Impact:** Table of contents and sidebar are more hierarchal, less of a laundry list

The image displays two side-by-side screenshots of the Chapel Documentation website, illustrating the changes in content categorization. Both screenshots show the 'Chapel Documentation' header with the version '1.26' and a search bar.

**Left Screenshot (Old Structure):**

- COMPILING AND RUNNING CHAPEL**
  - Quickstart Instructions
  - Using Chapel
  - Platform-Specific Notes
  - Technical Notes
  - Tools
  - Docs for Contributors
- WRITING CHAPEL PROGRAMS**
  - Quick Reference
  - Hello World Variants
  - Primers
- Language Specification**
  - Introductory Material
  - Language Basics
  - Code Structures
  - Composite Types
    - Tuples
    - Classes
    - Records
    - Unions
    - Ranges
    - Domains
    - Arrays
  - Generic Programming
  - Parallel Programming
  - Distributed Programming
  - Additional Topics

**Right Screenshot (New Structure):**

- COMPILING AND RUNNING CHAPEL**
  - Quickstart Instructions
  - Using Chapel
  - Platform-Specific Notes
  - Technical Notes
  - Tools
  - Docs for Contributors
- WRITING CHAPEL PROGRAMS**
  - Quick Reference
  - Hello World Variants
  - Primers
  - Language Specification
  - Built-in Types and Functions
  - Standard Modules
- Package Modules**
  - Algorithms
  - Communication (Inter-Locale)
  - Data Structures
    - ConcurrentMap
    - DistributedBag
    - DistributedDeque
    - DistributedFilters
    - LinkedLists
    - LockFreeQueue
    - LockFreeStack
    - SortedMap
    - SortedSet
    - UnrolledLinkedList
  - File Formats and I/O
  - Math / Numerical Computing
  - Memory Management
  - Networking / Inter-Process Communication

**Chapel Language Specification (New Structure):**

- Introductory Material**
  - Scope
  - Notation
  - Organization
  - Acknowledgments
  - Language Overview
- Language Basics**
  - Lexical Structure
  - Types
  - Variables
  - Conversions
  - Expressions
  - Statements
  - Input and Output
- Code Structures**
  - Modules
  - Procedures
  - Iterators
  - Methods
  - Error Handling
- Composite Types**
  - Tuples
  - Classes
  - Records
  - Unions
  - Ranges

**Package Modules (New Structure):**

- Package modules are libraries that currently live outside of the Chapel Standard Library, either because they are not considered to be fundamental enough or because they are not yet mature enough for inclusion there. Over time, we expect many of these to become `mason` packages.
- Algorithms**
  - Crypto
  - Search
  - Sort
- Communication (Inter-Locale)**
  - AllLocalesBarriers
  - CopyAggregation
  - MPI
  - UnorderedAtoms
  - UnorderedCopy
- Data Structures**
  - ConcurrentMap
  - DistributedBag
  - DistributedDeque
  - DistributedFilters
  - LinkedLists
  - LockFreeQueue
  - LockFreeStack
  - SortedMap
  - SortedSet
  - UnrolledLinkedList
- File Formats and I/O**
  - HDFS
  - HDFS



# DOCUMENTATION UPDATES

## Documentation Index

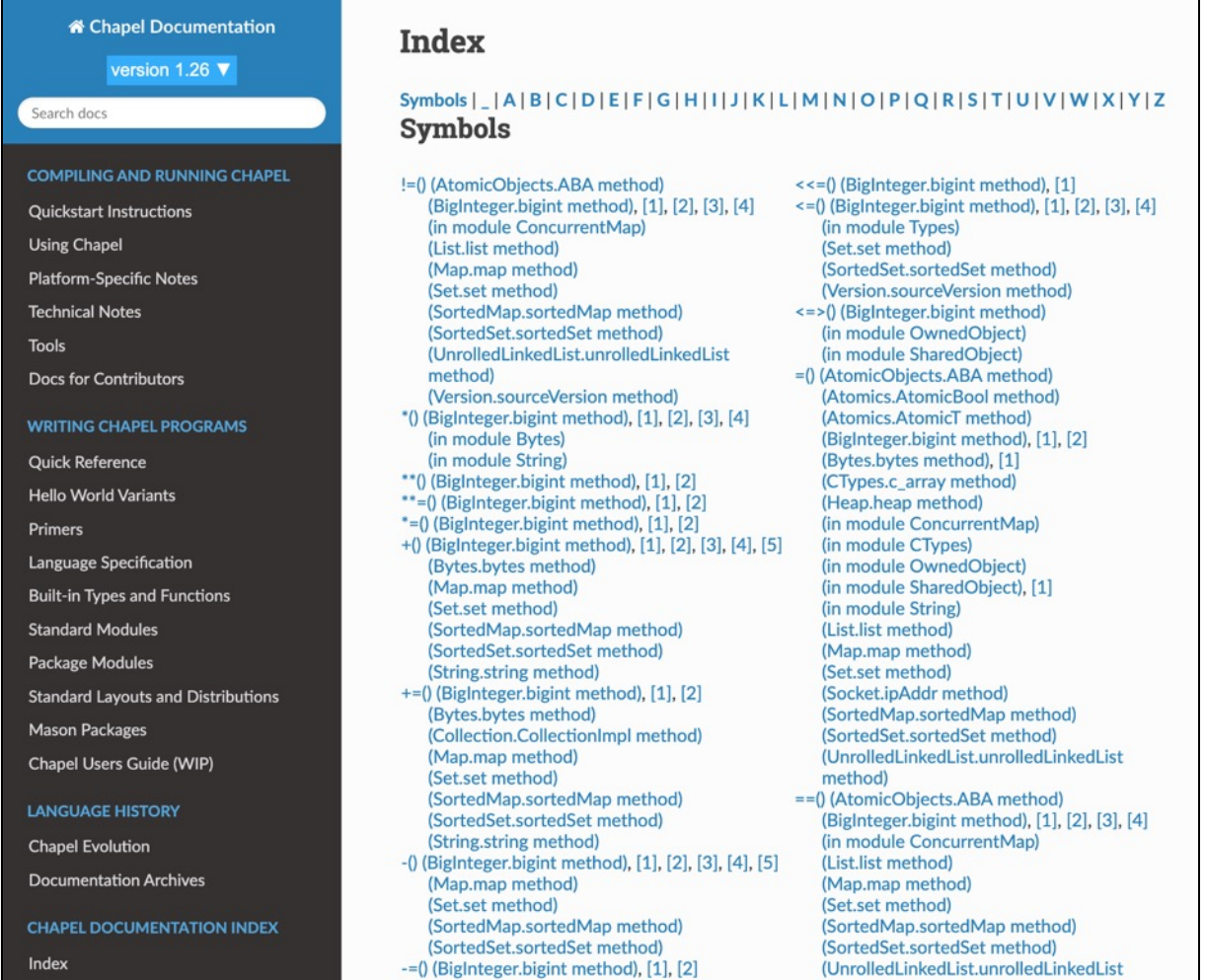
**Background:** Online docs have an index, but it's been somewhat difficult to find

- linked at the bottom of some pages, but not most

**This Effort:** Added it to the sidebar

### Next Steps:

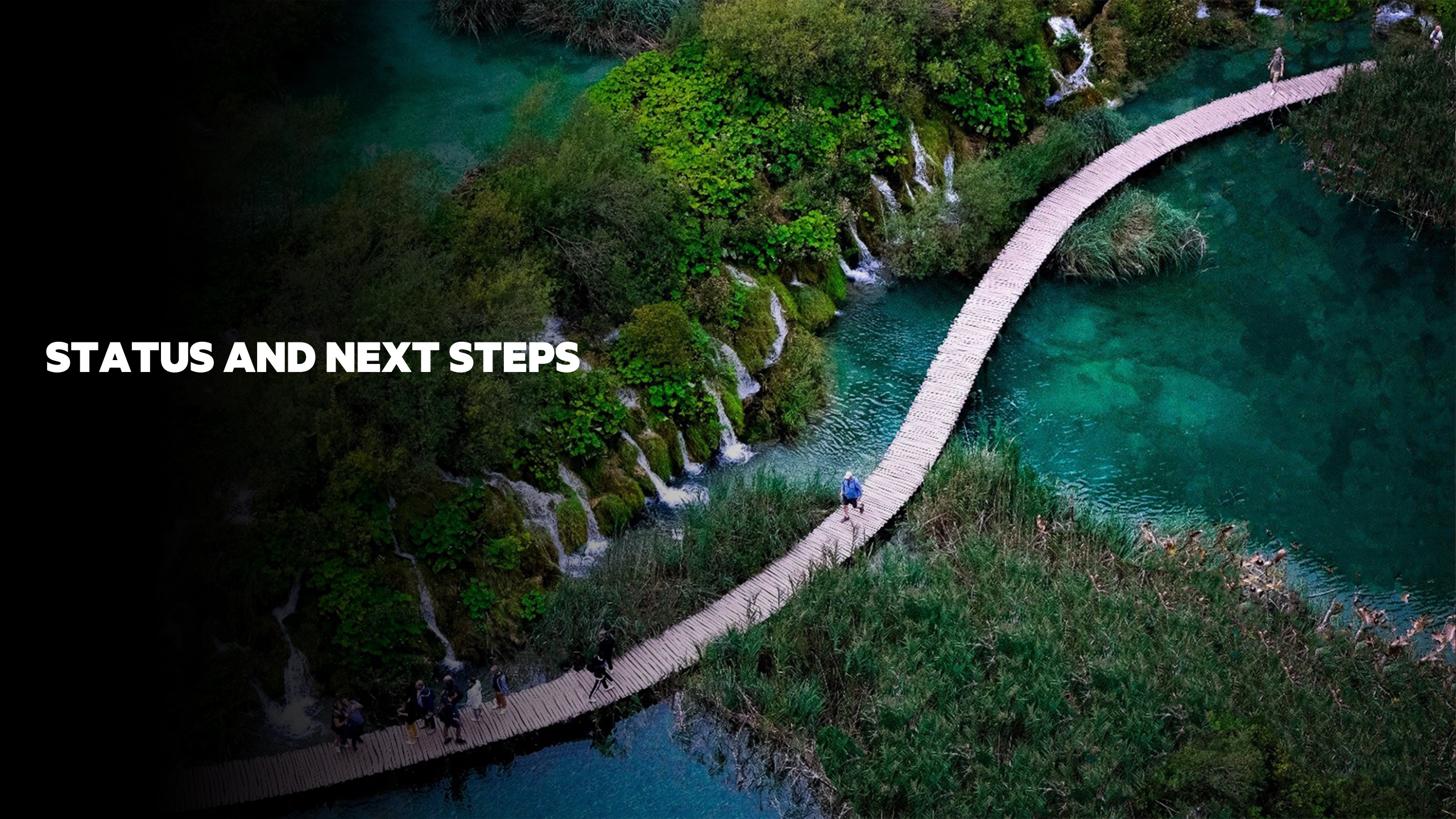
- Improve quality, utility of index
- Restore index entries to the language specification
- Support distinct indexes for spec, modules, etc.?



The screenshot shows the Chapel Documentation website. The top navigation bar includes "Chapel Documentation" and "version 1.26". A search bar is present. The sidebar on the left is organized into sections: "COMPILING AND RUNNING CHAPEL" (Quickstart Instructions, Using Chapel, Platform-Specific Notes, Technical Notes, Tools, Docs for Contributors), "WRITING CHAPEL PROGRAMS" (Quick Reference, Hello World Variants, Primers, Language Specification, Built-in Types and Functions, Standard Modules, Package Modules, Standard Layouts and Distributions, Mason Packages, Chapel Users Guide (WIP)), "LANGUAGE HISTORY" (Chapel Evolution, Documentation Archives), and "CHAPEL DOCUMENTATION INDEX" (Index). The main content area is titled "Index" and features a navigation bar with letters A-Z. Below this, a list of symbols and their corresponding methods is displayed, such as `!={()}` (AtomicObjects.ABA method), `<=>()` (BigInteger.bigint method), and `+=()` (BigInteger.bigint method).



# STATUS AND NEXT STEPS



# DOCUMENTATION UPDATES

## Status and Next Steps

---

### **Status:**

- Documentation continues to improve in organization, quality, utility

### **Next Steps:**

- Merge remaining “built-in types and functions” sections into the language specification
- Continue improving and refining contributor documentation and indexes



# OTHER DOCUMENTATION IMPROVEMENTS



## OTHER DOCUMENTATION IMPROVEMENTS

---

For a more complete list of documentation changes and improvements in the 1.25.1 and 1.26.0 releases, refer to the following sections in the [CHANGES.md](#) file:

- ‘Documentation’
- ‘Example Codes’



An aerial photograph showing a long, narrow wooden boardwalk that curves through a vibrant green landscape. The boardwalk is built over a river and several small waterfalls. Several people are seen walking along the boardwalk. The water is a clear, bright green color. The surrounding vegetation is dense and lush, with various shades of green.

# THANK YOU

---

<https://chapel-lang.org>  
@ChapelLanguage

