

**draft proposed American National Standard for
Information Systems—Programming
Language—Common Lisp**

X3J13/94-101R

Version 15.17R, X3J13/94-101R.
Fri 12-Aug-1994 6:35pm EDT

Versions 15.17 (X3J13/94-101) and 15.17R (X3J13/94-101R) are absolutely identical except for changes to the document number and date on the cover and in each page's header, addition of this disclaimer on the back of the cover page, additions to the Edit and Review History on page Credits iv, and changes to the names and headings in the credits section on pages Credits v-vii. There are NO technical changes between these versions; from a normative point of view, the documents X3J13/94-101 and X3J13/94-101R are entirely interchangeable.

Figures

Chapter 1 (Introduction)	
1-1. Notations for NIL	1-14
1-2. Deprecated Functions	1-33
1-3. Functions with Deprecated :TEST-NOT Arguments	1-33
1-4. Symbols in the COMMON-LISP package (part one of twelve).	1-35
1-5. Symbols in the COMMON-LISP package (part two of twelve).	1-36
1-6. Symbols in the COMMON-LISP package (part three of twelve).	1-37
1-7. Symbols in the COMMON-LISP package (part four of twelve).	1-38
1-8. Symbols in the COMMON-LISP package (part five of twelve).	1-39
1-9. Symbols in the COMMON-LISP package (part six of twelve).	1-40
1-10. Symbols in the COMMON-LISP package (part seven of twelve).	1-41
1-11. Symbols in the COMMON-LISP package (part eight of twelve).	1-42
1-12. Symbols in the COMMON-LISP package (part nine of twelve).	1-43
1-13. Symbols in the COMMON-LISP package (part ten of twelve).	1-44
1-14. Symbols in the COMMON-LISP package (part eleven of twelve).	1-45
1-15. Symbols in the COMMON-LISP package (part twelve of twelve).	1-46
Chapter 2 (Syntax)	
2-1. Readtable defined names	2-1
2-2. Variables that influence the Lisp reader.	2-2
2-3. Standard Character Subrepertoire (Part 1 of 3: Latin Characters)	2-3
2-4. Standard Character Subrepertoire (Part 2 of 3: Numeric Characters)	2-3
2-5. Standard Character Subrepertoire (Part 3 of 3: Special Characters)	2-4
2-6. Possible Character Syntax Types	2-5
2-7. Character Syntax Types in Standard Syntax	2-5
2-8. Constituent Traits of Standard Characters and Semi-Standard Characters	2-7
2-9. Syntax for Numeric Tokens	2-14
2-10. Examples of reserved tokens	2-15
2-11. Examples of symbols	2-15
2-12. Examples of symbols or potential numbers	2-16
2-13. Examples of Ratios	2-17
2-14. Examples of Floating-point numbers	2-18
2-15. Examples of the printed representation of symbols (Part 1 of 2)	2-19
2-16. Examples of the printed representation of symbols (Part 2 of 2)	2-20
2-17. Valid patterns for tokens	2-21
2-18. Examples of the use of double-quote	2-26

2-19. Standard # Dispatching Macro Character Syntax	2-30
2-20. Radix Indicator Example	2-34
2-21. Complex Number Example	2-35

Chapter 3 (Evaluation and Compilation)

3-1. Some Defined Names Applicable to Variables	3-4
3-2. Common Lisp Special Operators	3-6
3-3. Defined names applicable to macros	3-7
3-4. Some function-related defined names	3-8
3-5. Some operators applicable to receiving multiple values	3-13
3-6. Defined names applicable to compiler macros	3-16
3-7. EVAL-WHEN processing	3-21
3-8. Defining Macros That Affect the Compile-Time Environment	3-22
3-9. Common Lisp Declaration Identifiers	3-29
3-10. What Kind of Lambda Lists to Use	3-33
3-11. Defined names applicable to lambda lists	3-33
3-12. Standardized Operators that use Ordinary Lambda Lists	3-34
3-13. Lambda List Keywords used by Ordinary Lambda Lists	3-34
3-14. Lambda List Keywords used by Generic Function Lambda Lists	3-40
3-15. Standardized Operators that use Specialized Lambda Lists	3-40
3-16. Lambda List Keywords used by Specialized Lambda Lists	3-40
3-17. Operators that use Macro Lambda Lists	3-41
3-18. Lambda List Keywords used by Macro Lambda Lists	3-42
3-19. Lambda List Keywords used by Defsetf Lambda Lists	3-47
3-20. Lambda List Keywords used by Define-modify-macro Lambda Lists	3-48
3-21. Lambda List Keywords used by Define-method-combination arguments Lambda Lists	3-48
3-22. Global Declaration Specifiers	3-82
3-23. Standardized Forms In Which Declarations Can Occur	3-84
3-24. Local Declaration Specifiers	3-84
3-25. Optimize qualities	3-97

Chapter 4 (Types and Classes)

4-1. Cross-References to Data Type Information	4-2
4-2. Standardized Atomic Type Specifiers	4-4
4-3. Standardized Compound Type Specifier Names	4-5
4-4. Standardized Compound-Only Type Specifier Names	4-5
4-5. Defined names relating to types and declarations.	4-6
4-6. Standardized Type Specifier Names	4-7
4-7. Object System Classes	4-8
4-8. Classes that correspond to pre-defined type specifiers	4-18
4-9. Result possibilities for subtypep	4-35

Chapter 5 (Data and Control Flow)

5-1. Examples of setf	5-1
5-2. Operators relating to places and generalized reference.	5-1
5-3. Sample Setf Expansion of a Variable	5-4
5-4. Sample Setf Expansion of a CAR Form	5-4
5-5. Sample Setf Expansion of a SUBSEQ Form	5-4
5-6. Sample Setf Expansion of a LDB Form	5-5
5-7. Functions that setf can be used with—1	5-6
5-8. Functions that setf can be used with—2	5-7
5-9. Read-Modify-Write Macros	5-12
5-10. Macros that have implicit tagbodies.	5-49
5-11. Operators that always prefer EQ over EQL	5-57
5-12. Summary and priorities of behavior of equal	5-61
5-13. Summary and priorities of behavior of equalp	5-63

Chapter 6 (Iteration)

Chapter 7 (Objects)

7-1. Standardized Method-Defining Operators	7-18
7-2. Built-in Method Combination Types	7-26

Chapter 8 (Structures)

Chapter 9 (Conditions)

9-1. Standardized Condition Types	9-2
9-2. Operators that define and create conditions.	9-2
9-3. Operators that read condition slots.	9-3
9-4. Operators relating to handling conditions.	9-7
9-5. Defined names relating to signaling conditions.	9-7
9-6. Defined names relating to restarts.	9-9
9-7. Operators relating to assertions.	9-10

Chapter 10 (Symbols)

10-1. Property list defined names	10-1
10-2. Symbol creation and inquiry defined names	10-1

Chapter 11 (Packages)

11-1. Some Defined Names related to Packages	11-1
11-2. Standardized Package Names	11-4

Chapter 12 (Numbers)

12-1. Operators relating to Arithmetic.	12-1
---	------

12-2. Defined names relating to Exponentials, Logarithms, and Trigonometry.	12-1
12-3. Operators for numeric comparison and predication.	12-1
12-4. Defined names relating to numeric type manipulation and coercion.	12-2
12-5. Defined names relating to logical operations on numbers.	12-3
12-6. Defined names relating to byte manipulation.	12-4
12-7. Defined names relating to implementation-dependent details about numbers.	12-4
12-8. Functions Affected by Rule of Float Substitutability	12-6
12-9. Trigonometric Identities for Complex Domain	12-9
12-10. Quadrant Numbering for Branch Cuts	12-9
12-11. Random-state defined names	12-10
12-12. Recommended Minimum Floating-Point Precision and Exponent Size	12-14
12-13. Uses of /=, =, <, >, <=, and >=	12-22
12-14. Mathematical definition of arc sine, arc cosine, and arc tangent	12-29
12-15. Quadrant information for arc tangent	12-31
12-16. Mathematical definitions for hyperbolic functions	12-33
12-17. Bit-Wise Logical Operations	12-68
12-18. Bit-wise Logical Operations on Integers	12-72
Chapter 13 (Characters)	
13-1. Character defined names - 1	13-1
13-2. Character defined names - 2	13-1
Chapter 14 (Conses)	
14-1. Some defined names relating to conses.	14-1
14-2. Some defined names relating to trees.	14-1
14-3. Some defined names relating to lists.	14-2
14-4. Some defined names related to association lists.	14-2
14-5. Some defined names related to sets.	14-3
14-6. CAR and CDR variants	14-11
Chapter 15 (Arrays)	
15-1. General Purpose Array-Related Defined Names	15-2
15-2. Operators that Manipulate Strings	15-3
15-3. Operators that Manipulate Bit Arrays	15-4
15-4. Bit-wise Logical Operations on Bit Arrays	15-38
Chapter 16 (Strings)	
Chapter 17 (Sequences)	
17-1. Standardized Sequence Functions	17-1
17-2. Operators that have Two-Argument Tests to be Satisfied	17-2
17-3. Operators that have One-Argument Tests to be Satisfied	17-3

Chapter 18 (Hash Tables)	
18-1. Hash-table defined names	18-1
Chapter 19 (Filenames)	
19-1. Pathname Operations	19-2
19-2. Pathname functions using a :CASE argument	19-4
19-3. Special Markers In Directory Component	19-8
Chapter 20 (Files)	
20-1. File and Directory Operations	20-1
20-2. File Functions that Treat Open and Closed Streams Differently	20-1
20-3. File Functions where Closed Streams Might Work Best	20-2
Chapter 21 (Streams)	
21-1. Some General-Purpose Stream Operations	21-1
21-2. Operators relating to Input Streams.	21-1
21-3. Operators relating to Output Streams.	21-2
21-4. Operators relating to Bidirectional Streams.	21-2
21-5. Defined Names related to Specialized Streams	21-4
21-6. Standardized Stream Variables	21-4
21-7. Operators that accept either Open or Closed Streams	21-5
21-8. Operators that accept Open Streams only	21-5
Chapter 22 (Printer)	
22-1. Standardized Printer Control Variables	22-1
22-2. Additional Influences on the Lisp printer.	22-2
22-3. Example of Logical Blocks, Conditional Newlines, and Sections	22-15
22-4. Defined names related to pretty printing.	22-15
22-5. Format directives related to Pretty Printing	22-15
22-6. Examples of format control strings	22-23
22-7. Argument correspondences for the WRITE function.	22-63
Chapter 23 (Reader)	
23-1. Values of standard control variables	23-18
Chapter 24 (System Construction)	
24-1. Features examples	24-2
Chapter 25 (Environment)	
25-1. Variables maintained by the Read-Eval-Print Loop	25-1
25-2. Defined names relating to debugging	25-1

25-3. Defined names relating to environment inquiry.	25-2
25-4. Defined names involving Time.	25-2
25-5. Defined names involving time in Decoded Time.	25-3
25-6. Defined names involving time in Universal Time.	25-4
25-7. Defined names involving time in Internal Time.	25-4
25-8. Defined names involving time in Seconds.	25-4

Chapter 26 (Glossary)

26-1. Exponent Markers	26-22
26-2. Standardized I/O Customization Variables	26-30
26-3. Standardized Iteration Forms	26-33
26-4. Standardized Restart Functions	26-49

Chapter A (Appendix)

Contents

Chapter 1 (Introduction)	
1.1 Scope, Purpose, and History	1-1
1.1.1 Scope and Purpose	1-1
1.1.2 History	1-1
1.2 Organization of the Document	1-4
1.3 Referenced Publications	1-5
1.4 Definitions	1-7
1.4.1 Notational Conventions	1-7
1.4.1.1 Font Key	1-7
1.4.1.2 Modified BNF Syntax	1-7
1.4.1.2.1 Splicing in Modified BNF Syntax	1-8
1.4.1.2.2 Indirection in Modified BNF Syntax	1-9
1.4.1.2.3 Additional Uses for Indirect Definitions in Modified BNF Syntax	1-10
1.4.1.3 Special Symbols	1-10
1.4.1.4 Objects with Multiple Notations	1-12
1.4.1.4.1 Case in Symbols	1-12
1.4.1.4.2 Numbers	1-13
1.4.1.4.3 Use of the Dot Character	1-13
1.4.1.4.4 NIL	1-13
1.4.1.5 Designators	1-14
1.4.1.6 Nonsense Words	1-15
1.4.2 Error Terminology	1-15
1.4.3 Sections Not Formally Part Of This Standard	1-18
1.4.4 Interpreting Dictionary Entries	1-19
1.4.4.1 The “Affected By” Section of a Dictionary Entry	1-19
1.4.4.2 The “Arguments” Section of a Dictionary Entry	1-19
1.4.4.3 The “Arguments and Values” Section of a Dictionary Entry	1-19
1.4.4.4 The “Binding Types Affected” Section of a Dictionary Entry	1-19
1.4.4.5 The “Class Precedence List” Section of a Dictionary Entry	1-19
1.4.4.6 Dictionary Entries for Type Specifiers	1-20
1.4.4.6.1 The “Compound Type Specifier Kind” Section of a Dictionary Entry	1-20
1.4.4.6.2 The “Compound Type Specifier Syntax” Section of a Dictionary Entry	1-20
1.4.4.6.3 The “Compound Type Specifier Arguments” Section of a Dictionary Entry	1-21
1.4.4.6.4 The “Compound Type Specifier Description” Section of a Dictionary Entry	1-21
1.4.4.7 The “Constant Value” Section of a Dictionary Entry	1-21
1.4.4.8 The “Description” Section of a Dictionary Entry	1-21
1.4.4.9 The “Examples” Section of a Dictionary Entry	1-21

1.4.4.10 The “Exceptional Situations” Section of a Dictionary Entry	1-21
1.4.4.11 The “Initial Value” Section of a Dictionary Entry	1-21
1.4.4.12 The “Argument Precedence Order” Section of a Dictionary Entry	1-22
1.4.4.13 The “Method Signature” Section of a Dictionary Entry	1-22
1.4.4.14 The “Name” Section of a Dictionary Entry	1-22
1.4.4.15 The “Notes” Section of a Dictionary Entry	1-24
1.4.4.16 The “Pronunciation” Section of a Dictionary Entry	1-24
1.4.4.17 The “See Also” Section of a Dictionary Entry	1-24
1.4.4.18 The “Side Effects” Section of a Dictionary Entry	1-24
1.4.4.19 The “Supertypes” Section of a Dictionary Entry	1-24
1.4.4.20 The “Syntax” Section of a Dictionary Entry	1-24
1.4.4.20.1 Special “Syntax” Notations for Overloaded Operators	1-25
1.4.4.20.2 Naming Conventions for Rest Parameters	1-25
1.4.4.20.3 Requiring Non-Null Rest Parameters in the “Syntax” Section	1-25
1.4.4.20.4 Return values in the “Syntax” Section	1-26
1.4.4.20.4.1 No Arguments or Values in the “Syntax” Section	1-26
1.4.4.20.4.2 Unconditional Transfer of Control in the “Syntax” Section	1-26
1.4.4.21 The “Valid Context” Section of a Dictionary Entry	1-26
1.4.4.22 The “Value Type” Section of a Dictionary Entry	1-26
1.5 Conformance	1-28
1.5.1 Conforming Implementations	1-28
1.5.1.1 Required Language Features	1-28
1.5.1.2 Documentation of Implementation-Dependent Features	1-28
1.5.1.3 Documentation of Extensions	1-28
1.5.1.4 Treatment of Exceptional Situations	1-28
1.5.1.4.1 Resolution of Apparent Conflicts in Exceptional Situations	1-28
1.5.1.4.1.1 Examples of Resolution of Apparent Conflicts in Exceptional Situations	1-28
1.5.1.5 Conformance Statement	1-29
1.5.2 Conforming Programs	1-29
1.5.2.1 Use of Implementation-Defined Language Features	1-29
1.5.2.1.1 Use of Read-Time Conditionals	1-30
1.5.2.2 Character Set for Portable Code	1-30
1.6 Language Extensions	1-31
1.7 Language Subsets	1-32
1.8 Deprecated Language Features	1-33
1.8.1 Deprecated Functions	1-33
1.8.2 Deprecated Argument Conventions	1-33
1.8.3 Deprecated Variables	1-33
1.8.4 Deprecated Reader Syntax	1-34
1.9 Symbols in the COMMON-LISP Package	1-35
Chapter 2 (Syntax)	
2.1 Character Syntax	2-1
2.1.1 Readtables	2-1
2.1.1.1 The Current Readtable	2-1

2.1.1.2 The Standard Readtable	2-1
2.1.1.3 The Initial Readtable	2-1
2.1.2 Variables that affect the Lisp Reader	2-2
2.1.3 Standard Characters	2-2
2.1.4 Character Syntax Types	2-4
2.1.4.1 Constituent Characters	2-6
2.1.4.2 Constituent Traits	2-6
2.1.4.3 Invalid Characters	2-8
2.1.4.4 Macro Characters	2-8
2.1.4.5 Multiple Escape Characters	2-8
2.1.4.5.1 Examples of Multiple Escape Characters	2-9
2.1.4.6 Single Escape Character	2-9
2.1.4.6.1 Examples of Single Escape Characters	2-9
2.1.4.7 Whitespace Characters	2-9
2.1.4.7.1 Examples of Whitespace Characters	2-9
2.2 Reader Algorithm	2-11
2.3 Interpretation of Tokens	2-14
2.3.1 Numbers as Tokens	2-14
2.3.1.1 Potential Numbers as Tokens	2-14
2.3.1.1.1 Escape Characters and Potential Numbers	2-15
2.3.1.1.2 Examples of Potential Numbers	2-15
2.3.2 Constructing Numbers from Tokens	2-16
2.3.2.1 Syntax of a Rational	2-16
2.3.2.1.1 Syntax of an Integer	2-16
2.3.2.1.2 Syntax of a Ratio	2-16
2.3.2.2 Syntax of a Float	2-17
2.3.2.3 Syntax of a Complex	2-18
2.3.3 The Consing Dot	2-18
2.3.4 Symbols as Tokens	2-18
2.3.5 Valid Patterns for Tokens	2-20
2.3.6 Package System Consistency Rules	2-22
2.4 Standard Macro Characters	2-23
2.4.1 Left-Parenthesis	2-23
2.4.2 Right-Parenthesis	2-23
2.4.3 Single-Quote	2-23
2.4.3.1 Examples of Single-Quote	2-24
2.4.4 Semicolon	2-24
2.4.4.1 Examples of Semicolon	2-24
2.4.4.2 Notes about Style for Semicolon	2-24
2.4.4.2.1 Use of Single Semicolon	2-24
2.4.4.2.2 Use of Double Semicolon	2-24
2.4.4.2.3 Use of Triple Semicolon	2-25
2.4.4.2.4 Use of Quadruple Semicolon	2-25
2.4.4.2.5 Examples of Style for Semicolon	2-25
2.4.5 Double-Quote	2-25

2.4.6 Backquote	2-26
2.4.6.1 Notes about Backquote	2-28
2.4.7 Comma	2-28
2.4.8 Sharpsign	2-28
2.4.8.1Sharpsign Backslash	2-31
2.4.8.2Sharpsign Single-Quote	2-31
2.4.8.3Sharpsign Left-Parenthesis	2-31
2.4.8.4 Sharpsign Asterisk	2-32
2.4.8.4.1 Examples of Sharpsign Asterisk	2-32
2.4.8.5 Sharpsign Colon	2-33
2.4.8.6 Sharpsign Dot	2-33
2.4.8.7 Sharpsign B	2-33
2.4.8.8 Sharpsign O	2-33
2.4.8.9 Sharpsign X	2-33
2.4.8.10 Sharpsign R	2-34
2.4.8.11 Sharpsign C	2-34
2.4.8.12 Sharpsign A	2-35
2.4.8.13 Sharpsign S	2-35
2.4.8.14Sharpsign P	2-36
2.4.8.15Sharpsign Equal-Sign	2-36
2.4.8.16Sharpsign Sharpsign	2-36
2.4.8.17 Sharpsign Plus	2-37
2.4.8.18 Sharpsign Minus	2-37
2.4.8.19 Sharpsign Vertical-Bar	2-37
2.4.8.19.1Examples of Sharpsign Vertical-Bar	2-38
2.4.8.19.2 Notes about Style for Sharpsign Vertical-Bar	2-39
2.4.8.20 Sharpsign Less-Than-Sign	2-39
2.4.8.21 Sharpsign Whitespace	2-39
2.4.8.22 Sharpsign Right-Parenthesis	2-39
2.4.9 Re-Reading Abbreviated Expressions	2-40
Chapter 3 (Evaluation and Compilation)	
3.1 Evaluation	3-1
3.1.1 Introduction to Environments	3-1
3.1.1.1 The Global Environment	3-1
3.1.1.2 Dynamic Environments	3-1
3.1.1.3 Lexical Environments	3-2
3.1.1.3.1 The Null Lexical Environment	3-2
3.1.1.4 Environment Objects	3-3
3.1.2 The Evaluation Model	3-3
3.1.2.1 Form Evaluation	3-3
3.1.2.1.1 Symbols as Forms	3-3
3.1.2.1.1.1 Lexical Variables	3-4
3.1.2.1.1.2 Dynamic Variables	3-4
3.1.2.1.1.3 Constant Variables	3-5

3.1.2.1.1.4 Symbols Naming Both Lexical and Dynamic Variables	3-5
3.1.2.1.2 Conses as Forms	3-5
3.1.2.1.2.1 Special Forms	3-6
3.1.2.1.2.2 Macro Forms	3-6
3.1.2.1.2.3 Function Forms	3-7
3.1.2.1.2.4 Lambda Forms	3-8
3.1.2.1.3 Self-Evaluating Objects	3-8
3.1.2.1.3.1 Examples of Self-Evaluating Objects	3-9
3.1.3 Lambda Expressions	3-9
3.1.4 Closures and Lexical Binding	3-9
3.1.5 Shadowing	3-11
3.1.6 Extent	3-12
3.1.7 Return Values	3-13
3.2 Compilation	3-14
3.2.1 Compiler Terminology	3-14
3.2.2 Compilation Semantics	3-15
3.2.2.1 Compiler Macros	3-15
3.2.2.1.1 Purpose of Compiler Macros	3-16
3.2.2.1.2 Naming of Compiler Macros	3-16
3.2.2.1.3 When Compiler Macros Are Used	3-17
3.2.2.1.3.1 Notes about the Implementation of Compiler Macros	3-17
3.2.2.2 Minimal Compilation	3-17
3.2.2.3 Semantic Constraints	3-18
3.2.3 File Compilation	3-19
3.2.3.1 Processing of Top Level Forms	3-20
3.2.3.1.1 Processing of Defining Macros	3-22
3.2.3.1.2 Constraints on Macros and Compiler Macros	3-22
3.2.4 Literal Objects in Compiled Files	3-23
3.2.4.1 Externalizable Objects	3-23
3.2.4.2 Similarity of Literal Objects	3-24
3.2.4.2.1 Similarity of Aggregate Objects	3-24
3.2.4.2.2 Definition of Similarity	3-24
3.2.4.3 Extensions to Similarity Rules	3-26
3.2.4.4 Additional Constraints on Externalizable Objects	3-26
3.2.5 Exceptional Situations in the Compiler	3-28
3.3 Declarations	3-29
3.3.1 Minimal Declaration Processing Requirements	3-29
3.3.2 Declaration Specifiers	3-29
3.3.3 Declaration Identifiers	3-29
3.3.3.1 Shorthand notation for Type Declarations	3-30
3.3.4 Declaration Scope	3-30
3.3.4.1 Examples of Declaration Scope	3-30
3.4 Lambda Lists	3-33
3.4.1 Ordinary Lambda Lists	3-33
3.4.1.1 Specifiers for the required parameters	3-34

3.4.1.2	Specifiers for optional parameters	3-35
3.4.1.3	A specifier for a rest parameter	3-35
3.4.1.4	Specifiers for keyword parameters	3-35
3.4.1.4.1	Suppressing Keyword Argument Checking	3-36
3.4.1.4.1.1	Examples of Suppressing Keyword Argument Checking	3-37
3.4.1.5	Specifiers for &aux variables	3-37
3.4.1.6	Examples of Ordinary Lambda Lists	3-37
3.4.2	Generic Function Lambda Lists	3-39
3.4.3	Specialized Lambda Lists	3-40
3.4.4	Macro Lambda Lists	3-41
3.4.4.1	Destructuring by Lambda Lists	3-43
3.4.4.1.1	Data-directed Destructuring by Lambda Lists	3-43
3.4.4.1.1.1	Examples of Data-directed Destructuring by Lambda Lists	3-43
3.4.4.1.2	Lambda-list-directed Destructuring by Lambda Lists	3-43
3.4.5	Destructuring Lambda Lists	3-45
3.4.6	Boa Lambda Lists	3-45
3.4.7	Defsetf Lambda Lists	3-47
3.4.8	Deftype Lambda Lists	3-47
3.4.9	Define-modify-macro Lambda Lists	3-48
3.4.10	Define-method-combination Arguments Lambda Lists	3-48
3.4.11	Syntactic Interaction of Documentation Strings and Declarations	3-48
3.5	Error Checking in Function Calls	3-50
3.5.1	Argument Mismatch Detection	3-50
3.5.1.1	Safe and Unsafe Calls	3-50
3.5.1.1.1	Error Detection Time in Safe Calls	3-51
3.5.1.2	Too Few Arguments	3-51
3.5.1.3	Too Many Arguments	3-51
3.5.1.4	Unrecognized Keyword Arguments	3-51
3.5.1.5	Invalid Keyword Arguments	3-51
3.5.1.6	Odd Number of Keyword Arguments	3-52
3.5.1.7	Destructuring Mismatch	3-52
3.5.1.8	Errors When Calling a Next Method	3-52
3.6	Traversal Rules and Side Effects	3-53
3.7	Destructive Operations	3-54
3.7.1	Modification of Literal Objects	3-54
3.7.2	Transfer of Control during a Destructive Operation	3-55
3.7.2.1	Examples of Transfer of Control during a Destructive Operation	3-55
3.8	Evaluation and Compilation Dictionary	3-56
Chapter 4 (Types and Classes)		
4.1	Introduction	4-1
4.2	Types	4-2
4.2.1	Data Type Definition	4-2
4.2.2	Type Relationships	4-2
4.2.3	Type Specifiers	4-3

4.3 Classes	4-8
4.3.1 Introduction to Classes	4-8
4.3.1.1 Standard Metaclasses	4-9
4.3.2 Defining Classes	4-9
4.3.3 Creating Instances of Classes	4-10
4.3.4 Inheritance	4-11
4.3.4.1 Examples of Inheritance	4-11
4.3.4.2 Inheritance of Class Options	4-11
4.3.5 Determining the Class Precedence List	4-11
4.3.5.1 Topological Sorting	4-12
4.3.5.2 Examples of Class Precedence List Determination	4-13
4.3.6 Redefining Classes	4-14
4.3.6.1 Modifying the Structure of Instances	4-15
4.3.6.2 Initializing Newly Added Local Slots	4-15
4.3.6.3 Customizing Class Redefinition	4-16
4.3.7 Integrating Types and Classes	4-16
4.4 Types and Classes Dictionary	4-19
Chapter 5 (Data and Control Flow)	
5.1 Generalized Reference	5-1
5.1.1 Overview of Places and Generalized Reference	5-1
5.1.1.1 Evaluation of Subforms to Places	5-1
5.1.1.1.1 Examples of Evaluation of Subforms to Places	5-2
5.1.1.2 Setf Expansions	5-3
5.1.1.2.1 Examples of Setf Expansions	5-4
5.1.2 Kinds of Places	5-5
5.1.2.1 Variable Names as Places	5-5
5.1.2.2 Function Call Forms as Places	5-5
5.1.2.3 VALUES Forms as Places	5-9
5.1.2.4 THE Forms as Places	5-9
5.1.2.5 APPLY Forms as Places	5-9
5.1.2.6 Setf Expansions and Places	5-10
5.1.2.7 Macro Forms as Places	5-10
5.1.2.8 Symbol Macros as Places	5-10
5.1.2.9 Other Compound Forms as Places	5-10
5.1.3 Treatment of Other Macros Based on SETF	5-11
5.2 Transfer of Control to an Exit Point	5-13
5.3 Data and Control Flow Dictionary	5-14
Chapter 6 (Iteration)	
6.1 The LOOP Facility	6-1
6.1.1 Overview of the Loop Facility	6-1
6.1.1.1 Simple vs Extended Loop	6-1
6.1.1.1.1 Simple Loop	6-1

6.1.1.1.2Extended Loop	6-1
6.1.1.2Loop Keywords	6-1
6.1.1.3Parsing Loop Clauses	6-1
6.1.1.4 Expanding Loop Forms	6-2
6.1.1.5 Summary of Loop Clauses	6-3
6.1.1.5.1 Summary of Variable Initialization and Stepping Clauses	6-3
6.1.1.5.2 Summary of Value Accumulation Clauses	6-3
6.1.1.5.3 Summary of Termination Test Clauses	6-4
6.1.1.5.4 Summary of Unconditional Execution Clauses	6-4
6.1.1.5.5 Summary of Conditional Execution Clauses	6-5
6.1.1.5.6 Summary of Miscellaneous Clauses	6-5
6.1.1.6 Order of Execution	6-5
6.1.1.7 Destructuring	6-6
6.1.1.8 Restrictions on Side-Effects	6-8
6.1.2 Variable Initialization and Stepping Clauses	6-8
6.1.2.1 Iteration Control	6-8
6.1.2.1.1 The for-as-arithmetic subclause	6-9
6.1.2.1.1.1Examples of for-as-arithmetic subclause	6-10
6.1.2.1.2The for-as-in-list subclause	6-11
6.1.2.1.2.1Examples of for-as-in-list subclause	6-11
6.1.2.1.3 The for-as-on-list subclause	6-12
6.1.2.1.3.1 Examples of for-as-on-list subclause	6-12
6.1.2.1.4 The for-as-equals-then subclause	6-12
6.1.2.1.4.1 Examples of for-as-equals-then subclause	6-12
6.1.2.1.5 The for-as-across subclause	6-12
6.1.2.1.5.1 Examples of for-as-across subclause	6-13
6.1.2.1.6 The for-as-hash subclause	6-13
6.1.2.1.7 The for-as-package subclause	6-14
6.1.2.1.7.1 Examples of for-as-package subclause	6-15
6.1.2.2 Local Variable Initializations	6-15
6.1.2.2.1Examples of WITH clause	6-16
6.1.3 Value Accumulation Clauses	6-17
6.1.3.1Examples of COLLECT clause	6-19
6.1.3.2Examples of APPEND and NCONC clauses	6-19
6.1.3.3Examples of COUNT clause	6-20
6.1.3.4Examples of MAXIMIZE and MINIMIZE clauses	6-20
6.1.3.5Examples of SUM clause	6-20
6.1.4 Termination Test Clauses	6-21
6.1.4.1Examples of REPEAT clause	6-22
6.1.4.2Examples of ALWAYS, NEVER, and THEREIS clauses	6-22
6.1.4.3Examples of WHILE and UNTIL clauses	6-24
6.1.5Unconditional Execution Clauses	6-24
6.1.5.1Examples of unconditional execution	6-24
6.1.6Conditional Execution Clauses	6-25
6.1.6.1Examples of WHEN clause	6-25

6.1.7 Miscellaneous Clauses	6-26
6.1.7.1 Control Transfer Clauses	6-27
6.1.7.1.1 Examples of NAMED clause	6-27
6.1.7.2 Initial and Final Execution	6-27
6.1.8 Examples of Miscellaneous Loop Features	6-27
6.1.8.1 Examples of clause grouping	6-28
6.1.9 Notes about Loop	6-30
6.2 Iteration Dictionary	6-31
Chapter 7 (Objects)	
7.1 Object Creation and Initialization	7-1
7.1.1 Initialization Arguments	7-2
7.1.2 Declaring the Validity of Initialization Arguments	7-2
7.1.3 Defaulting of Initialization Arguments	7-3
7.1.4 Rules for Initialization Arguments	7-4
7.1.5 Shared-Initialize	7-5
7.1.6 Initialize-Instance	7-6
7.1.7 Definitions of Make-Instance and Initialize-Instance	7-7
7.2 Changing the Class of an Instance	7-9
7.2.1 Modifying the Structure of the Instance	7-9
7.2.2 Initializing Newly Added Local Slots	7-9
7.2.3 Customizing the Change of Class of an Instance	7-10
7.3 Reinitializing an Instance	7-11
7.3.1 Customizing Reinitialization	7-11
7.4 Meta-Objects	7-12
7.4.1 Standard Meta-objects	7-12
7.5 Slots	7-13
7.5.1 Introduction to Slots	7-13
7.5.2 Accessing Slots	7-13
7.5.3 Inheritance of Slots and Slot Options	7-14
7.6 Generic Functions and Methods	7-17
7.6.1 Introduction to Generic Functions	7-17
7.6.2 Introduction to Methods	7-18
7.6.3 Agreement on Parameter Specializers and Qualifiers	7-20
7.6.4 Congruent Lambda-lists for all Methods of a Generic Function	7-20
7.6.5 Keyword Arguments in Generic Functions and Methods	7-21
7.6.5.1 Examples of Keyword Arguments in Generic Functions and Methods	7-21
7.6.6 Method Selection and Combination	7-22
7.6.6.1 Determining the Effective Method	7-22
7.6.6.1.1 Selecting the Applicable Methods	7-22
7.6.6.1.2 Sorting the Applicable Methods by Precedence Order	7-23
7.6.6.1.3 Applying method combination to the sorted list of applicable methods	7-23
7.6.6.2 Standard Method Combination	7-24
7.6.6.3 Declarative Method Combination	7-25
7.6.6.4 Built-in Method Combination Types	7-26

7.6.7 Inheritance of Methods	7-27
7.7 Objects Dictionary	7-28
Chapter 8 (Structures)	
8.1 Structures Dictionary	8-1
Chapter 9 (Conditions)	
9.1 Condition System Concepts	9-1
9.1.1 Condition Types	9-2
9.1.1.1 Serious Conditions	9-3
9.1.2 Creating Conditions	9-3
9.1.2.1 Condition Designators	9-3
9.1.3 Printing Conditions	9-4
9.1.3.1 Recommended Style in Condition Reporting	9-4
9.1.3.1.1 Capitalization and Punctuation in Condition Reports	9-5
9.1.3.1.2 Leading and Trailing Newlines in Condition Reports	9-5
9.1.3.1.3 Embedded Newlines in Condition Reports	9-5
9.1.3.1.4 Note about Tabs in Condition Reports	9-6
9.1.3.1.5 Mentioning Containing Function in Condition Reports	9-6
9.1.4 Signaling and Handling Conditions	9-6
9.1.4.1 Signaling	9-7
9.1.4.1.1 Resignaling a Condition	9-8
9.1.4.2 Restarts	9-8
9.1.4.2.1 Interactive Use of Restarts	9-9
9.1.4.2.2 Interfaces to Restarts	9-9
9.1.4.2.3 Restart Tests	9-9
9.1.4.2.4 Associating a Restart with a Condition	9-9
9.1.5 Assertions	9-10
9.1.6 Notes about the Condition System's Background	9-10
9.2 Conditions Dictionary	9-11
Chapter 10 (Symbols)	
10.1 Symbol Concepts	10-1
10.2 Symbols Dictionary	10-2
Chapter 11 (Packages)	
11.1 Package Concepts	11-1
11.1.1 Introduction to Packages	11-1
11.1.1.1 Package Names and Nicknames	11-1
11.1.1.2 Symbols in a Package	11-1
11.1.1.2.1 Internal and External Symbols	11-1
11.1.1.2.2 Package Inheritance	11-2
11.1.1.2.3 Accessibility of Symbols in a Package	11-2

11.1.1.2.4 Locating a Symbol in a Package	11-3
11.1.1.2.5 Prevention of Name Conflicts in Packages	11-3
11.1.2 Standardized Packages	11-4
11.1.2.1 The COMMON-LISP Package	11-4
11.1.2.1.1 Constraints on the COMMON-LISP Package for Conforming Implementations	11-5
11.1.2.1.2 Constraints on the COMMON-LISP Package for Conforming Programs	11-5
11.1.2.1.2.1 Some Exceptions to Constraints on the COMMON-LISP Package for Conforming Programs	11-6
11.1.2.2 The COMMON-LISP-USER Package	11-6
11.1.2.3 The KEYWORD Package	11-7
11.1.2.3.1 Interning a Symbol in the KEYWORD Package	11-7
11.1.2.3.2 Notes about The KEYWORD Package	11-7
11.1.2.4 Implementation-Defined Packages	11-7
11.2 Packages Dictionary	11-8
Chapter 12 (Numbers)	
12.1 Number Concepts	12-1
12.1.1 Numeric Operations	12-1
12.1.1.1 Associativity and Commutativity in Numeric Operations	12-2
12.1.1.1.1 Examples of Associativity and Commutativity in Numeric Operations	12-2
12.1.1.1.2 Contagion in Numeric Operations	12-3
12.1.1.1.3 Viewing Integers as Bits and Bytes	12-3
12.1.1.1.3.1 Logical Operations on Integers	12-3
12.1.1.1.3.2 Byte Operations on Integers	12-3
12.1.2 Implementation-Dependent Numeric Constants	12-4
12.1.3 Rational Computations	12-4
12.1.3.1 Rule of Unbounded Rational Precision	12-4
12.1.3.2 Rule of Canonical Representation for Rationals	12-5
12.1.3.3 Rule of Float Substitutability	12-5
12.1.4 Floating-point Computations	12-6
12.1.4.1 Rule of Float and Rational Contagion	12-6
12.1.4.1.1 Examples of Rule of Float and Rational Contagion	12-6
12.1.4.2 Rule of Float Approximation	12-7
12.1.4.3 Rule of Float Underflow and Overflow	12-7
12.1.4.4 Rule of Float Precision Contagion	12-7
12.1.5 Complex Computations	12-7
12.1.5.1 Rule of Complex Substitutability	12-8
12.1.5.2 Rule of Complex Contagion	12-8
12.1.5.3 Rule of Canonical Representation for Complex Rationals	12-8
12.1.5.3.1 Examples of Rule of Canonical Representation for Complex Rationals	12-8
12.1.5.4 Principal Values and Branch Cuts	12-8
12.1.6 Interval Designators	12-9
12.1.7 Random-State Operations	12-10
12.2 Numbers Dictionary	12-11

Chapter 13 (Characters)	
13.1 Character Concepts	13-1
13.1.1 Introduction to Characters	13-1
13.1.2 Introduction to Scripts and Repertoires	13-1
13.1.2.1 Character Scripts	13-1
13.1.2.2 Character Repertoires	13-2
13.1.3 Character Attributes	13-2
13.1.4 Character Categories	13-2
13.1.4.1 Graphic Characters	13-3
13.1.4.2 Alphabetic Characters	13-3
13.1.4.3 Characters With Case	13-3
13.1.4.3.1 Uppercase Characters	13-3
13.1.4.3.2 Lowercase Characters	13-4
13.1.4.3.3 Corresponding Characters in the Other Case	13-4
13.1.4.3.4 Case of Implementation-Defined Characters	13-4
13.1.4.4 Numeric Characters	13-4
13.1.4.5 Alphanumeric Characters	13-4
13.1.4.6 Digits in a Radix	13-4
13.1.5 Identity of Characters	13-5
13.1.6 Ordering of Characters	13-5
13.1.7 Character Names	13-5
13.1.8 Treatment of Newline during Input and Output	13-6
13.1.9 Character Encodings	13-7
13.1.10 Documentation of Implementation-Defined Scripts	13-7
13.2 Characters Dictionary	13-8
Chapter 14 (Conses)	
14.1 Cons Concepts	14-1
14.1.1 Conses as Trees	14-1
14.1.1.1 General Restrictions on Parameters that must be Trees	14-1
14.1.2 Conses as Lists	14-1
14.1.2.1 Lists as Association Lists	14-2
14.1.2.2 Lists as Sets	14-2
14.1.2.3 General Restrictions on Parameters that must be Lists	14-3
14.2 Conses Dictionary	14-4
Chapter 15 (Arrays)	
15.1 Array Concepts	15-1
15.1.1 Array Elements	15-1
15.1.1.1 Array Indices	15-1
15.1.1.2 Array Dimensions	15-1
15.1.1.2.1 Implementation Limits on Individual Array Dimensions	15-1
15.1.1.3 Array Rank	15-1
15.1.1.3.1 Vectors	15-1

15.1.1.3.1.1 Fill Pointers	15-1
15.1.1.3.2 Multidimensional Arrays	15-2
15.1.1.3.2.1 Storage Layout for Multidimensional Arrays	15-2
15.1.1.3.2.2 Implementation Limits on Array Rank	15-2
15.1.2 Specialized Arrays	15-2
15.1.2.1 Array Upgrading	15-3
15.1.2.2 Required Kinds of Specialized Arrays	15-3
15.2 Arrays Dictionary	15-5
Chapter 16 (Strings)	
16.1 String Concepts	16-1
16.1.1 Implications of Strings Being Arrays	16-1
16.1.2 Subtypes of STRING	16-1
16.2 Strings Dictionary	16-2
Chapter 17 (Sequences)	
17.1 Sequence Concepts	17-1
17.1.1 General Restrictions on Parameters that must be Sequences	17-1
17.2 Rules about Test Functions	17-2
17.2.1 Satisfying a Two-Argument Test	17-2
17.2.1.1 Examples of Satisfying a Two-Argument Test	17-2
17.2.2 Satisfying a One-Argument Test	17-3
17.2.2.1 Examples of Satisfying a One-Argument Test	17-4
17.3 Sequences Dictionary	17-5
Chapter 18 (Hash Tables)	
18.1 Hash Table Concepts	18-1
18.1.1 Hash-Table Operations	18-1
18.1.2 Modifying Hash Table Keys	18-1
18.1.2.1 Visible Modification of Objects with respect to EQ and EQL	18-2
18.1.2.2 Visible Modification of Objects with respect to EQUAL	18-2
18.1.2.2.1 Visible Modification of Conses with respect to EQUAL	18-2
18.1.2.2.2 Visible Modification of Bit Vectors and Strings with respect to EQUAL	18-2
18.1.2.3 Visible Modification of Objects with respect to EQUALP	18-2
18.1.2.3.1 Visible Modification of Structures with respect to EQUALP	18-2
18.1.2.3.2 Visible Modification of Arrays with respect to EQUALP	18-3
18.1.2.3.3 Visible Modification of Hash Tables with respect to EQUALP	18-3
18.1.2.4 Visible Modifications by Language Extensions	18-3
18.2 Hash Tables Dictionary	18-4
Chapter 19 (Filenames)	
19.1 Overview of Filenames	19-1
19.1.1 Namestrings as Filenames	19-1

19.1.2 Pathnames as Filenames	19-1
19.1.3 Parsing Namestrings Into Pathnames	19-2
19.2 Pathnames	19-3
19.2.1 Pathname Components	19-3
19.2.1.1 The Pathname Host Component	19-3
19.2.1.2 The Pathname Device Component	19-3
19.2.1.3 The Pathname Directory Component	19-3
19.2.1.4 The Pathname Name Component	19-3
19.2.1.5 The Pathname Type Component	19-3
19.2.1.6 The Pathname Version Component	19-3
19.2.2 Interpreting Pathname Component Values	19-3
19.2.2.1 Strings in Component Values	19-3
19.2.2.1.1 Special Characters in Pathname Components	19-4
19.2.2.1.2 Case in Pathname Components	19-4
19.2.2.1.2.1 Local Case in Pathname Components	19-4
19.2.2.1.2.2 Common Case in Pathname Components	19-4
19.2.2.2 Special Pathname Component Values	19-5
19.2.2.2.1 NIL as a Component Value	19-5
19.2.2.2.2 :WILD as a Component Value	19-5
19.2.2.2.3 :UNSPECIFIC as a Component Value	19-5
19.2.2.2.3.1 Relation between component values NIL and :UNSPECIFIC	19-6
19.2.2.3 Restrictions on Wildcard Pathnames	19-6
19.2.2.4 Restrictions on Examining Pathname Components	19-6
19.2.2.4.1 Restrictions on Examining a Pathname Host Component	19-7
19.2.2.4.2 Restrictions on Examining a Pathname Device Component	19-7
19.2.2.4.3 Restrictions on Examining a Pathname Directory Component	19-7
19.2.2.4.3.1 Directory Components in Non-Hierarchical File Systems	19-8
19.2.2.4.4 Restrictions on Examining a Pathname Name Component	19-9
19.2.2.4.5 Restrictions on Examining a Pathname Type Component	19-9
19.2.2.4.6 Restrictions on Examining a Pathname Version Component	19-9
19.2.2.4.7 Notes about the Pathname Version Component	19-9
19.2.2.5 Restrictions on Constructing Pathnames	19-9
19.2.3 Merging Pathnames	19-10
19.2.3.1 Examples of Merging Pathnames	19-10
19.3 Logical Pathnames	19-12
19.3.1 Syntax of Logical Pathname Namestrings	19-12
19.3.1.1 Additional Information about Parsing Logical Pathname Namestrings	19-12
19.3.1.1.1 The Host part of a Logical Pathname Namestring	19-13
19.3.1.1.2 The Device part of a Logical Pathname Namestring	19-13
19.3.1.1.3 The Directory part of a Logical Pathname Namestring	19-13
19.3.1.1.4 The Type part of a Logical Pathname Namestring	19-13
19.3.1.1.5 The Version part of a Logical Pathname Namestring	19-13
19.3.1.1.6 Wildcard Words in a Logical Pathname Namestring	19-13
19.3.1.1.7 Lowercase Letters in a Logical Pathname Namestring	19-13
19.3.1.1.8 Other Syntax in a Logical Pathname Namestring	19-13

19.3.2 Logical Pathname Components	19-14
19.3.2.1 Unspecific Components of a Logical Pathname	19-14
19.3.2.2 Null Strings as Components of a Logical Pathname	19-14
19.4 Filenames Dictionary	19-15
Chapter 20 (Files)	
20.1 File System Concepts	20-1
20.1.1 Coercion of Streams to Pathnames	20-1
20.1.2 File Operations on Open and Closed Streams	20-1
20.1.3 Truenames	20-2
20.1.3.1 Examples of Truenames	20-2
20.2 Files Dictionary	20-3
Chapter 21 (Streams)	
21.1 Stream Concepts	21-1
21.1.1 Introduction to Streams	21-1
21.1.1.1 Abstract Classifications of Streams	21-1
21.1.1.1.1 Input, Output, and Bidirectional Streams	21-1
21.1.1.1.2 Open and Closed Streams	21-2
21.1.1.1.3 Interactive Streams	21-2
21.1.1.2 Abstract Classifications of Streams	21-3
21.1.1.2.1 File Streams	21-3
21.1.1.3 Other Subclasses of Stream	21-3
21.1.2 Stream Variables	21-4
21.1.3 Stream Arguments to Standardized Functions	21-5
21.1.4 Restrictions on Composite Streams	21-6
21.2 Streams Dictionary	21-7
Chapter 22 (Printer)	
22.1 The Lisp Printer	22-1
22.1.1 Overview of The Lisp Printer	22-1
22.1.1.1 Multiple Possible Textual Representations	22-1
22.1.1.1.1 Printer Escaping	22-2
22.1.2 Printer Dispatching	22-2
22.1.3 Default Print-Object Methods	22-2
22.1.3.1 Printing Numbers	22-2
22.1.3.1.1 Printing Integers	22-2
22.1.3.1.2 Printing Ratios	22-3
22.1.3.1.3 Printing Floats	22-3
22.1.3.1.4 Printing Complexes	22-3
22.1.3.1.5 Note about Printing Numbers	22-4
22.1.3.2 Printing Characters	22-4
22.1.3.3 Printing Symbols	22-4

22.1.3.3.1 Package Prefixes for Symbols	22-5
22.1.3.3.2 Effect of Readtable Case on the Lisp Printer	22-5
22.1.3.3.2.1 Examples of Effect of Readtable Case on the Lisp Printer	22-6
22.1.3.4 Printing Strings	22-8
22.1.3.5 Printing Lists and Conses	22-8
22.1.3.6 Printing Bit Vectors	22-9
22.1.3.7 Printing Other Vectors	22-9
22.1.3.8 Printing Other Arrays	22-10
22.1.3.9 Examples of Printing Arrays	22-11
22.1.3.10 Printing Random States	22-11
22.1.3.11 Printing Pathnames	22-11
22.1.3.12 Printing Structures	22-11
22.1.3.13 Printing Other Objects	22-12
22.1.4 Examples of Printer Behavior	22-12
22.2 The Lisp Pretty Printer	22-14
22.2.1 Pretty Printer Concepts	22-14
22.2.1.1 Dynamic Control of the Arrangement of Output	22-14
22.2.1.2 Format Directive Interface	22-15
22.2.1.3 Compiling Format Strings	22-16
22.2.1.4 Pretty Print Dispatch Tables	22-16
22.2.1.5 Pretty Printer Margins	22-16
22.2.2 Examples of using the Pretty Printer	22-16
22.2.3 Notes about the Pretty Printer's Background	22-22
22.3 Formatted Output	22-23
22.3.1 FORMAT Basic Output	22-24
22.3.1.1 Tilde C: Character	22-24
22.3.1.2 Tilde Percent: Newline	22-25
22.3.1.3 Tilde Ampersand: Fresh-Line	22-25
22.3.1.4 Tilde Vertical-Bar: Page	22-25
22.3.1.5 Tilde Tilde: Tilde	22-25
22.3.2 FORMAT Radix Control	22-25
22.3.2.1 Tilde R: Radix	22-25
22.3.2.2 Tilde D: Decimal	22-26
22.3.2.3 Tilde B: Binary	22-26
22.3.2.4 Tilde O: Octal	22-26
22.3.2.5 Tilde X: Hexadecimal	22-27
22.3.3 FORMAT Floating-Point Printers	22-27
22.3.3.1 Tilde F: Fixed-Format Floating-Point	22-27
22.3.3.2 Tilde E: Exponential Floating-Point	22-28
22.3.3.3 Tilde G: General Floating-Point	22-30
22.3.3.4 Tilde Dollarsign: Monetary Floating-Point	22-30
22.3.4 FORMAT Printer Operations	22-31
22.3.4.1 Tilde A: Aesthetic	22-31
22.3.4.2 Tilde S: Standard	22-31
22.3.4.3 Tilde W: Write	22-31

22.3.5 FORMAT Pretty Printer Operations	22-32
22.3.5.1 Tilde Underscore: Conditional Newline	22-32
22.3.5.2 Tilde Less-Than-Sign: Logical Block	22-32
22.3.5.3 Tilde I: Indent	22-33
22.3.5.4 Tilde Slash: Call Function	22-33
22.3.6 FORMAT Layout Control	22-34
22.3.6.1 Tilde T: Tabulate	22-34
22.3.6.2 Tilde Less-Than-Sign: Justification	22-34
22.3.6.3 Tilde Greater-Than-Sign: End of Justification	22-36
22.3.7 FORMAT Control-Flow Operations	22-36
22.3.7.1 Tilde Asterisk: Go-To	22-36
22.3.7.2 Tilde Left-Bracket: Conditional Expression	22-36
22.3.7.3 Tilde Right-Bracket: End of Conditional Expression	22-37
22.3.7.4 Tilde Left-Brace: Iteration	22-37
22.3.7.5 Tilde Right-Brace: End of Iteration	22-38
22.3.7.6 Tilde Question-Mark: Recursive Processing	22-39
22.3.8 FORMAT Miscellaneous Operations	22-39
22.3.8.1 Tilde Left-Paren: Case Conversion	22-39
22.3.8.2 Tilde Right-Paren: End of Case Conversion	22-40
22.3.8.3 Tilde P: Plural	22-40
22.3.9 FORMAT Miscellaneous Pseudo-Operations	22-40
22.3.9.1 Tilde Semicolon: Clause Separator	22-40
22.3.9.2 Tilde Circumflex: Escape Upward	22-40
22.3.9.3 Tilde Newline: Ignored Newline	22-42
22.3.10 Additional Information about FORMAT Operations	22-42
22.3.10.1 Nesting of FORMAT Operations	22-42
22.3.10.2 Missing and Additional FORMAT Arguments	22-43
22.3.10.3 Additional FORMAT Parameters	22-43
22.3.10.4 Undefined FORMAT Modifier Combinations	22-43
22.3.11 Examples of FORMAT	22-43
22.3.12 Notes about FORMAT	22-45
22.4 Printer Dictionary	22-46
Chapter 23 (Reader)	
23.1 Reader Concepts	23-1
23.1.1 Dynamic Control of the Lisp Reader	23-1
23.1.2 Effect of Readtable Case on the Lisp Reader	23-1
23.1.2.1 Examples of Effect of Readtable Case on the Lisp Reader	23-1
23.1.3 Argument Conventions of Some Reader Functions	23-2
23.1.3.1 The EOF-ERROR-P argument	23-2
23.1.3.2 The RECURSIVE-P argument	23-2
23.2 Reader Dictionary	23-4

Chapter 24 (System Construction)

24.1 System Construction Concepts	24-1
24.1.1 Loading	24-1
24.1.2 Features	24-1
24.1.2.1 Feature Expressions	24-1
24.1.2.1.1 Examples of Feature Expressions	24-2
24.2 System Construction Dictionary	24-3
Chapter 25 (Environment)	
25.1 The External Environment	25-1
25.1.1 Top level loop	25-1
25.1.2 Debugging Utilities	25-1
25.1.3 Environment Inquiry	25-1
25.1.4 Time	25-2
25.1.4.1 Decoded Time	25-2
25.1.4.2 Universal Time	25-3
25.1.4.3 Internal Time	25-4
25.1.4.4 Seconds	25-4
25.2 Environment Dictionary	25-5
Chapter 26 (Glossary)	
26.1 Glossary	26-1
Chapter A (Appendix)	
A.1 Removed Language Features	A-1
A.1.1 Requirements for removed and deprecated features	A-1
A.1.2 Removed Types	A-1
A.1.3 Removed Operators	A-1
A.1.4 Removed Argument Conventions	A-1
A.1.5 Removed Variables	A-1
A.1.6 Removed Reader Syntax	A-1
A.1.7 Packages No Longer Required	A-1

Index

- # 2-29
- &allow-other-keys 3-35
- &aux 3-37
- &body 3-42
- &environment 3-42
- &key 3-35
- &optional 3-35
- &rest 3-35
- &whole 3-42
- ' 2-24
- (2-23
- () 1-13, 26-2
- (setf class-name) 7-89
- (setf documentation) 25-17
-) 2-23
- * 12-35, 25-26
- ** 25-26
- *** 25-26
- *break-on-signals* 9-35
- *break-on-warnings* A-1
- *compile-file-pathname* 24-12
- *compile-file-truename* 24-12
- *compile-print* 24-13
- *compile-verbose* 24-13
- *debug-io* 21-57
- *debugger-hook* 9-34
- *default-pathname-defaults* 19-28
- *error-output* 21-57
- *features* 1-30, 2-37, 24-10
- *gensym-counter* 10-9
- *load-pathname* 24-13
- *load-print* 24-14
- *load-truename* 24-13
- *load-verbose* 24-14
- *macroexpand-hook* 3-80
- *modules* 24-14
- *package* 11-40
- *print-array* 22-66
- *print-base* 22-67
- *print-case* 22-68
- *print-circle* 2-36, 2-37, 22-69
- *print-escape* 22-70
- *print-gensym* 22-71
- *print-length* 22-72
- *print-level* 22-72
- *print-lines* 22-73
- *print-miser-width* 22-74
- *print-pprint-dispatch* 22-75
- *print-pretty* 22-75
- *print-radix* 22-67
- *print-readably* 22-76
- *print-right-margin* 22-78
- *query-io* 21-57
- *random-state* 12-53
- *read-base* 2-33, 2-34, 23-19
- *read-default-float-format* 23-19
- *read-eval* 2-33, 23-20
- *read-suppress* 23-21
- *readtable* 23-22
- *standard-input* 21-57
- *standard-output* 21-57
- *terminal-io* 21-59
- *trace-output* 21-57
- + 7-26, 12-36, 25-25
- ++ 25-25
- +++ 25-25
- , 2-28
- 12-36, 25-24
- . 2-23
- .. 2-40, 22-74
- ... 2-40, 22-55
- / 12-37, 25-27
- // 25-27
- /// 25-27
- /= 12-20
- 1+ 12-38
- 1- 12-38
- :absolute 19-7
- :back 19-8
- :common 19-5
- :compile-toplevel 3-60
- :execute 3-60

<code>:load-toplevel</code> 3-60	<code>alphanumericp</code> 13-15
<code>:local</code> 19-4	<code>ampersand</code> 26-4
<code>:relative</code> 19-7	Ampersand (format directive) 22-25
<code>:unspecific</code> 19-6	<code>and</code> 4-28, 5-68, 7-26
<code>:up</code> 19-8	<code>anonymous</code> 26-4
<code>:wild</code> 19-5, 19-8	<code>apparently uninterned</code> 26-4
<code>:wild-inferiors</code> 19-5, 19-8	<code>append</code> 7-26, 14-30
<code>;</code> 2-24	<code>applicable</code> 26-4
<code><</code> 12-20	<code>applicable handler</code> 26-4
<code><=</code> 12-20	<code>applicable method</code> 26-4
<code>=</code> 12-20	<code>applicable restart</code> 26-4
<code>></code> 12-20	<code>apply</code> 26-4
<code>>=</code> 12-20	<code>apply</code> 5-14
A (format directive) 22-31	<code>apropos</code> 25-8
A (sharpsign reader macro) 2-35	<code>apropos-list</code> 25-8
<code>abort</code> 9-65, 9-68	<code>aref</code> 15-18
<code>abs</code> 12-39	<code>argument</code> 26-4
<code>absolute</code> 26-2	<code>argument evaluation order</code> 26-4
<code>:absolute</code> 19-7	<code>argument precedence order</code> 26-4
<code>access</code> 26-2	<code>arithmetic-error</code> 12-89
<code>accessibility</code> 26-2	<code>arithmetic-error-operands</code> 12-89
<code>accessible</code> 11-2, 26-2	<code>arithmetic-error-operation</code> 12-89
<code>accessor</code> 26-2	<code>around method</code> 26-4
<code>acons</code> 14-43	<code>array</code> 26-4
<code>acos</code> 12-29	<code>array</code> 2-35, 15-5
<code>acosh</code> 12-33	<code>array element type</code> 26-5
<code>active</code> 15-2, 26-3	<code>array total size</code> 26-5
<code>actual adjustability</code> 26-3	<code>array-dimension</code> 15-19
<code>actual argument</code> 26-3	<code>array-dimension-limit</code> 15-30
<code>actual array element type</code> 15-3, 26-3	<code>array-dimensions</code> 15-20
<code>actual complex part type</code> 26-3	<code>array-displacement</code> 15-22
<code>actual parameter</code> 26-3	<code>array-element-type</code> 15-21
<code>actually adjustable</code> 26-3	<code>array-has-fill-pointer-p</code> 15-21
<code>add-method</code> 7-87	<code>array-in-bounds-p</code> 15-23
<code>adjoin</code> 14-54	<code>array-rank</code> 15-24
<code>adjust-array</code> 15-14	<code>array-rank-limit</code> 15-30
<code>adjustability</code> 26-3	<code>array-row-major-index</code> 15-25
<code>adjustable</code> 26-3	<code>array-total-size</code> 15-26
<code>adjustable-array-p</code> 15-18	<code>array-total-size-limit</code> 15-31
<code>after method</code> 26-3	<code>arrayp</code> 15-27
<code>alist</code> 26-3	<code>ash</code> 12-64
<code>allocate-instance</code> 7-30	<code>asin</code> 12-29
<code>alpha-char-p</code> 13-14	<code>asinh</code> 12-33
<code>alphabetic</code> 26-3	<code>assert</code> 9-15
<code>alphanumeric</code> 26-3	<code>assign</code> 26-5

assoc 14-43
assoc-if 14-43
assoc-if-not 14-43
association list 14-2, 26-5
asterisk 26-5
Asterisk (format directive) 22-36
Asterisk (sharpsign reader macro) 2-32
at-sign 26-5
atan 12-29
atanh 12-33
atom 26-5
atom 14-5, 14-7
atomic 26-5
atomic type specifier 26-5
attribute 26-5
aux variable 26-5
auxiliary method 26-5
B (format directive) 22-26
B (sharpsign reader macro) 2-33
:back 19-8
backquote 26-6
Backquote (reader macro) 2-26
backslash 26-6
Backslash (sharpsign reader macro) 2-31
bar 1-15
base character 26-6
base string 26-6
base-char 13-8
base-string 16-2
baz 1-15
before method 26-6
bidirectional 21-2, 26-6
bignum 12-20
binary 21-1, 26-6
bind 26-6
binding 3-1, 26-6
bit 26-6
bit 12-19, 15-36
bit array 26-6
bit vector 15-3, 26-6
bit-and 15-37
bit-andc1 15-37
bit-andc2 15-37
bit-equiv 15-37
bit-ior 15-37
bit-nand 15-37
bit-nor 15-37
bit-not 15-37
bit-orc1 15-37
bit-orc2 15-37
bit-vector 2-32, 15-9
bit-vector-p 15-39
bit-wise logical operation specifier 26-6
bit-xor 15-37
block 26-7
block 5-42
block tag 26-7
bnf key 1-8
boa lambda list 3-46, 26-7
body parameter 26-7
boole 12-67
boole-1 12-70
boole-2 12-70
boole-and 12-70
boole-andc1 12-70
boole-andc2 12-70
boole-c1 12-70
boole-c2 12-70
boole-clr 12-70
boole-equiv 12-70
boole-ior 12-70
boole-nand 12-70
boole-nor 12-70
boole-orc1 12-70
boole-orc2 12-70
boole-set 12-70
boole-xor 12-70
boolean 26-7
boolean 4-19
boolean equivalent 26-7
both-case-p 13-20
bound 26-7
bound declaration 3-30, 26-7
bounded 26-7
bounding index 26-7
bounding index designator 26-8
boundp 10-20
break 9-33
break loop 26-8
break-on-signals 9-35

broadcast stream 26-8
broadcast-stream 21-7
broadcast-stream-streams 21-47
built-in class 26-8
built-in type 26-8
built-in-class 4-23
butlast 14-33
byte 26-8
byte 12-76
byte specifier 26-8
byte-position 12-76
byte-size 12-76
C (format directive) 22-24
C (sharpsign reader macro) 2-34
caaaar 14-9
caaadr 14-9
caaar 14-9
caadar 14-9
caaddr 14-9
caadr 14-9
caar 14-9
cadaar 14-9
cadadr 14-9
cadar 14-9
caddar 14-9
caddr 14-9
cadr 26-8
cadr 14-9
call 26-8
call-arguments-limit 5-29
call-method 7-73
call-next-method 7-74
captured initialization form 26-8
car 26-9
car 14-9
case 26-9
case 5-74
case in symbol names 1-12
case sensitivity mode 26-9
catch 26-9
catch 5-43
catch tag 26-9
ccase 5-74
cdaaar 14-9
cdaadr 14-9
cdaar 14-9
cdadar 14-9
cdaddr 14-9
cdadr 14-9
cdar 14-9
cddaar 14-9
cddadr 14-9
cddar 14-9
cdddar 14-9
cddddr 14-9
cdddr 14-9
cddr 26-9
cddr 14-9
cdr 26-9
cdr 14-9
ceiling 12-25
cell 26-9
cell-error 9-13
cell-error-name 9-14
cerror 9-19
change-class 7-38
char 16-5
char-bit A-1
char-bits A-1
char-bits-limit A-1
char-code 13-21
char-code-limit 13-23
char-control-bit A-1
char-downcase 13-19
char-equal 13-10
char-font A-1
char-font-limit A-1
char-greaterp 13-10
char-hyper-bit A-1
char-int 13-22
char-lessp 13-10
char-meta-bit A-1
char-name 13-24
char-not-equal 13-10
char-not-greaterp 13-10
char-not-lessp 13-10
char-super-bit A-1
char-upcase 13-19
char/= 13-10

-
- char<** 13-10
 - char<=** 13-10
 - char=** 13-10
 - char>** 13-10
 - char>=** 13-10
 - character* 13-1, 21-1, 26-9
 - character** 2-31, 13-8, 13-12
 - character code* 26-9
 - character designator* 26-10
 - characterp** 13-13
 - check-type** 9-22
 - circular* 26-10
 - circular list* 14-2, 14-4, 26-10
 - Circumflex (format directive) 22-41
 - cis** 12-55
 - CL package* 11-4
 - CL-USER package* 11-7
 - class* 4-8, 26-10
 - class** 4-23
 - class designator* 26-10
 - class precedence list* 4-8, 4-12, 26-10
 - class-name** 7-88
 - class-of** 7-90
 - clear-input** 21-42
 - clear-output** 21-43
 - close* 26-10
 - close** 21-39
 - closed* 21-2, 26-10
 - closure* 26-10
 - clrhash** 18-14
 - coalesce* 3-14, 26-10
 - code* 26-11
 - code-char** A-1, 13-22
 - coerce* 26-11
 - coerce** 4-30
 - colon* 26-11
 - Colon (sharpsign reader macro) 2-33
 - comma* 26-11
 - Comma (reader macro) 2-28
 - comment* 2-24, 2-38
 - :common** 19-5
 - COMMON-LISP package* 1-35, 11-4
 - COMMON-LISP-USER package* 11-7
 - commonp** A-1
 - compilation* 26-11
 - compilation environment* 3-14, 26-11
 - compilation unit* 26-11
 - compilation-speed** 3-97
 - compile* 26-11
 - compile** 3-18, 3-57, 3-60
 - compile time* 3-15, 26-11
 - compile-file** 3-18, 24-3
 - compile-file-pathname** 24-5
 - *compile-file-pathname*** 24-12
 - *compile-file-truename*** 24-12
 - *compile-print*** 24-13
 - compile-time definition* 3-15, 26-11
 - :compile-toplevel** 3-60
 - *compile-verbose*** 24-13
 - compiled code* 3-14, 24-1, 26-11
 - compiled file* 24-1, 26-12
 - compiled function* 26-12
 - compiled-function** 4-22
 - compiled-function-p** 5-28
 - compiler* 3-14, 26-12
 - compiler macro* 3-18, 26-12
 - compiler macro expansion* 26-12
 - compiler macro form* 26-12
 - compiler macro function* 26-12
 - compiler-macro** 25-18
 - compiler-macro-function** 3-66
 - complement** 5-65
 - complex* 26-12
 - complex** 2-34, 12-11, 12-56, 22-4
 - complex float* 26-12
 - complex part type* 26-12
 - complex rational* 26-12
 - complex single float* 26-13
 - complexp** 12-57
 - composite stream* 26-13
 - compound form* 26-13
 - compound type specifier* 4-4, 26-13
 - compute-applicable-methods** 7-76
 - compute-restarts** 9-49
 - concatenate** 17-29
 - concatenated stream* 26-13
 - concatenated-stream** 21-8
 - concatenated-stream-streams** 21-50
 - cond** 5-69
 - condition* 26-13

- condition** 9–11
- condition designator* 9–3, 26–13
- condition handler* 26–13
- condition reporter* 9–4, 26–13
- conditional newline* 26–13
- conditional newlines* 22–14
- conformance* 26–13
- conforming code* 1–29, 26–13
- conforming implementation* 1–28, 26–13
- conforming processor* 26–14
- conforming program* 1–29, 26–14
- congruence* 7–19
- congruent* 26–14
- conjugate** 12–57
- cons* 14–1, 26–14
- cons** 2–26, 2–28, 14–5, 14–6
- consequences 1–17
- consp** 14–6
- constant* 26–14
- constant form* 26–14
- constant object* 26–14
- constant variable* 26–14
- constantly** 5–66
- constantp** 3–104
- constituent* 26–14
- constituent trait* 26–14
- constructed stream* 26–15
- contagion* 26–15
- continuable* 26–15
- continue** 9–66, 9–68
- control form* 26–15
- control-error** 5–104
- copy* 26–15
- copy-alist** 14–45
- copy-list** 14–19
- copy-pprint-dispatch** 22–46
- copy-readtable** 23–4
- copy-seq** 17–5
- copy-structure** 8–18
- copy-symbol** 10–7
- copy-tree** 14–12
- correctable* 26–15
- cos** 12–28
- cosh** 12–33
- count** 17–15
- count-if** 17–15
- count-if-not** 17–15
- ctypcase** 5–76
- current input base* 23–19, 26–15
- current logical block* 26–16
- current output base* 22–67, 26–16
- current package* 11–1, 26–16
- current pprint dispatch table* 22–16, 26–16
- current random state* 26–16
- current readtable* 2–1, 26–16
- D (format directive) 22–26
- data type* 26–16
- debug** 3–97
- debug I/O* 26–16
- *debug-io*** 21–57
- debugger* 26–16
- *debugger-hook*** 9–34
- decf** 12–43
- declaim** 3–83
- declaration* 3–29, 26–16
- declaration** 3–29, 3–96
- declaration identifier* 3–29, 26–16
- declaration specifier* 3–29, 26–16
- declare** 26–17
- declare** 3–83
- decline* 26–17
- decode-float** 12–82
- decode-universal-time** 25–5
- decoded time* 25–2, 26–17
- default method* 7–19, 26–17
- *default-pathname-defaults*** 19–28
- defaulted initialization argument list* 26–17
- defclass** 7–60
- defconstant** 5–31
- defgeneric** 7–65
- define-compiler-macro** 3–67
- define-condition** 9–42
- define-method-combination** 7–76
- define-method-combination arguments lambda list* 3–48, 26–17
- define-modify-macro** 5–92
- define-modify-macro lambda list* 3–48, 26–17
- define-setf-expander** 5–96
- define-symbol-macro** 3–77
- defined name* 26–17

defining form 26-17
defmacro 3-70
defmethod 7-69
defpackage 11-29
defparameter 5-32
defsetf 5-93
defsetf lambda list 3-47, 26-17
defstruct 8-1
deftype 4-33
deftype lambda list 3-48, 26-17
defun 5-15
defvar 5-32
delete 17-32
delete-duplicates 17-35
delete-file 20-9
delete-if 17-32
delete-if-not 17-32
delete-package 11-17
denominator 12-61
denormalized 26-17
deposit-field 12-77
derived type 26-18
derived type specifier 4-5, 26-18
describe 25-9
describe-object 25-10
designator 1-14, 26-18
destructive 26-18
destructuring lambda list 3-45, 26-18
destructuring-bind 5-35
different 26-18
digit 26-18
digit-char A-1, 13-16
digit-char-p 13-16
dimension 15-1, 26-18
direct instance 26-18
direct subclass 4-8, 26-18
direct superclass 4-8, 26-18
directory 20-3
directory-namestring 19-28
disassemble 25-17
disestablish 26-18
disjoint 26-19
dispatching macro character 26-19
displaced array 26-19
distinct 26-19
division-by-zero 12-90
do 6-31
do* 6-31
do-all-symbols 11-33
do-external-symbols 11-33
do-symbols 11-33
documentation 25-17
documentation string 26-19
dolist 6-37
Dollarsign (format directive) 22-30
dot 2-23, 22-55, 26-19
Dot (sharpsign reader macro) 2-33
Dot Dot 2-40, 22-74
Dot Dot Dot 2-40, 22-55
dotimes 6-35
dotted list 14-2, 14-4, 26-19
dotted pair 26-19
double float 26-19
double-float 12-14
double-float-epsilon 12-88
double-float-negative-epsilon 12-88
double-quote 26-19
Double-Quote (reader macro) 2-26
dpb 12-78
dribble 25-23
dynamic binding 26-19
dynamic environment 3-2, 26-19
dynamic extent 26-19
dynamic scope 26-20
dynamic variable 26-20
dynamic-extent 3-86
E (format directive) 22-28
ecase 5-74
echo stream 26-20
echo-stream 21-9
echo-stream-input-stream 21-49
echo-stream-output-stream 21-49
ed 25-22
effective method 7-22, 26-20
eighth 14-25
element 26-20
element type 26-20
elt 17-6
em 26-20
empty list 14-4, 26-20

empty type 26-20
encode-universal-time 25-6
end of file 26-20
end-of-file 21-61
endp 14-28
enough-namestring 19-28
ensure-directories-exist 20-4
ensure-generic-function 7-29
environment 3-1, 26-21
environment object 3-3, 26-21
environment parameter 26-21
eq 5-56
eql 4-30, 5-58
equal 5-59
Equal-Sign (sharpsign reader macro) 2-36
equalp 5-62
error 26-21
error 9-13, 9-17
error output 26-21
error terminology 1-15
error-output 21-57
escape 26-21
establish 26-21
etypecase 5-76
eval 2-33, 3-59, 3-60
eval-when 3-20, 3-60
evaluate 26-21
evaluation 3-1, 26-21
evaluation environment 3-14, 26-21
evaluation order 3-63, 5-2, 5-43, 5-83, 6-5, 6-9, 7-4, 9-16, 12-79
evenp 12-40
every 5-67
execute 26-22
:execute 3-60
execution time 26-22
exhaustive partition 26-22
exhaustive union 26-22
exit point 26-22
exp 12-41
explicit return 26-22
explicit use 26-22
exponent marker 26-22
export 26-22
export 11-8
exported 26-23
expressed adjustability 26-23
expressed array element type 15-3, 26-23
expressed complex part type 26-23
expression 26-23
expressly adjustable 26-23
expt 12-41
extended character 26-23
extended function designator 26-23
extended lambda list 26-23
extended-char 13-9
extension 26-24
extensions 1-17, 1-18
extent 26-24
external file format 26-24
external file format designator 26-24
external symbol 11-2, 26-24
externalizable object 3-23, 26-24
F (format directive) 22-27
false 26-24
fbound 26-24
fboundp 5-18
fceiling 12-25
fdefinition 5-17
feature 24-1, 26-24
feature expression 24-1, 26-25
features list 24-1, 26-25
features 1-30, 2-37, 24-10
ffloor 12-25
fifth 14-25
file 20-1, 26-25
file compiler 26-25
file position 26-25
file position designator 26-25
file stream 21-3, 26-25
file system 26-25
file-author 20-6
file-error 20-10
file-error-pathname 20-11
file-length 21-28
file-namestring 19-28
file-position 21-29
file-stream 21-9
file-string-length 21-31
file-write-date 20-7

filename 20-1, 26-25
fill 17-7
fill pointer 15-2, 26-25
fill-pointer 15-27
 fill-style conditional newline 22-18, 22-54
find 17-20
find-all-symbols 11-12
find-class 7-71
find-if 17-20
find-if-not 17-20
find-method 7-85
find-package 11-11
find-restart 9-50
find-symbol 11-9
finish-output 21-43
finite 26-25
first 14-25
fixnum 26-26
fixnum 12-20
flet 5-20
float 26-26
float 12-13, 12-85, 22-3
float-digits 12-82
float-precision 12-82
float-radix 12-82
float-sign 12-82
floating-point-inexact 12-91
floating-point-invalid-operation 12-90
floating-point-overflow 12-91
floating-point-underflow 12-91
floatp 12-86
floor 12-25
fmakunbound 5-19
 font key 1-7
foo 1-15
for-value 26-26
force-output 21-43
form 26-26
formal argument 26-26
formal parameter 26-26
format 26-26
format 22-80
format argument 26-26
format control 22-16, 26-26
format directive 26-26
format string 26-26
formatter 22-46
fourth 14-25
free declaration 3-30, 26-26
fresh 26-26
fresh-line 21-21
freshline 26-27
fround 12-25
ftruncate 12-25
ftype 3-95
funbound 26-27
funcall 5-24
function 26-27
function 2-31, 4-20, 5-25, 25-18
function block name 26-27
function cell 26-27
function designator 26-27
function form 26-27
function name 26-27
function-keywords 7-28
function-lambda-expression 5-26
functional evaluation 26-27
functional value 26-27
functionp 5-28
further compilation 3-14, 26-28
 G (format directive) 22-30
gcd 12-42
general 26-28
generalized boolean 26-28
generalized instance 26-28
generalized reference 5-1, 26-28
generalized synonym stream 26-28
generic function 4-22, 7-17, 26-28
generic function lambda list 3-39, 26-28
generic-function 4-22
gensym 26-28
gensym 10-8
gensym-counter 10-9
gentemp 10-10
get 10-17
get-decoded-time 25-6
get-dispatch-macro-character 23-13
get-internal-real-time 25-15
get-internal-run-time 25-16
get-macro-character 23-14

get-output-stream-string 21–52
get-properties 14–48
get-setf-expansion 5–98
get-universal-time 25–6
getf 14–49
gethash 18–10
global declaration 3–29, 26–29
global environment 3–1, 26–29
global variable 26–29
glyph 26–29
go 26–29
go 5–44
go point 26–29
go tag 26–29
graphic 13–3, 26–29
graphic-char-p 13–17
Greater-Than-Sign (format directive) 22–36
handle 26–29
handler 26–29
handler-bind 9–37
handler-case 9–38
hash table 26–29
hash-table 18–4
hash-table-count 18–6
hash-table-p 18–5
hash-table-rehash-size 18–7
hash-table-rehash-threshold 18–8
hash-table-size 18–9
hash-table-test 18–9
home package 26–29
host-namestring 19–28
I (format directive) 22–33
I/O customization variable 26–29
identical 26–30
identifier 26–30
identity 5–64
if 5–70
ignorable 3–85
ignore 3–85
ignore-errors 9–41
imagpart 12–59
immutable 26–30
implementation 26–30
implementation limit 26–30
implementation-defined 26–30
implementation-dependent 26–30
implementation-independent 26–30
implicit block 26–30
implicit compilation 3–14, 26–30
implicit progn 26–30
implicit tagbody 26–31
import 26–31
import 11–12
improper list 14–2, 26–31
in-package 11–26
inaccessible 26–31
incf 12–43
indefinite extent 26–31
indefinite scope 26–31
indicator 26–31
indirect instance 26–31
inherit 26–31
initial pprint dispatch table 26–31
initial readtable 2–2, 26–31
initialization argument list 7–1, 26–31
initialization form 26–31
initialize-instance 7–88
inline 3–93
input 21–1, 26–31
input-stream-p 21–11
inspect 25–23
instance 4–8, 26–31
int-char A–1
integer 26–31
integer 12–17
integer-decode-float 12–82
integer-length 12–65
integerp 12–66
interactive stream 21–2, 26–32
interactive-stream-p 21–12
intern 26–32
intern 11–35
internal symbol 11–2, 26–32
internal time 25–4, 26–32
internal time unit 26–32
internal-time-units-per-second 25–15
interned 26–32
interpreted function 26–32
interpreted implementation 26–32
intersection 14–52

-
- interval designator* 26-32
 - invalid* 26-32
 - invalid-method-error** 9-25
 - invoke-debugger** 9-32
 - invoke-restart** 9-52
 - invoke-restart-interactively** 9-53
 - is signaled 1-16
 - isqrt** 12-49
 - iteration form* 26-32
 - iteration variable* 26-33
 - key* 26-33
 - keyword* 26-33
 - KEYWORD package** 11-7
 - keyword** 10-4
 - keyword parameter* 26-33
 - keyword/value pair* 26-33
 - keywordp** 10-5
 - labels** 5-20
 - lambda** 3-56
 - lambda combination* 26-33
 - lambda expression* 26-33
 - lambda form* 26-33
 - lambda list* 3-33, 26-33
 - lambda list keyword* 26-34
 - lambda variable* 26-34
 - lambda-list-keywords** 5-30
 - lambda-parameters-limit** 5-30
 - last** 14-34
 - lcm** 12-44
 - ldb** 12-79
 - ldb-test** 12-80
 - ldiff** 14-35
 - leaf* 26-34
 - leap seconds* 26-34
 - least-negative-double-float** 12-87
 - least-negative-long-float** 12-87
 - least-negative-normalized-double-float** 12-87
 - least-negative-normalized-long-float** 12-87
 - least-negative-normalized-short-float** 12-87
 - least-negative-normalized-single-float** 12-87
 - least-negative-short-float** 12-87
 - least-negative-single-float** 12-87
 - least-positive-double-float** 12-87
 - least-positive-long-float** 12-87
 - least-positive-normalized-double-float** 12-87
 - least-positive-normalized-long-float** 12-87
 - least-positive-normalized-short-float** 12-87
 - least-positive-normalized-single-float** 12-87
 - least-positive-short-float** 12-87
 - least-positive-single-float** 12-87
 - leaves* 14-1
 - Left-Brace (format directive) 22-37
 - Left-Bracket (format directive) 22-36
 - Left-Paren (format directive) 22-39
 - left-parenthesis* 26-34
 - Left-Parenthesis (reader macro) 2-23
 - Left-Parenthesis (sharp-sign reader macro) 2-31
 - length* 26-34
 - length** 17-16
 - Less-Than-Sign (format directive) 22-32, 22-35
 - Less-Than-Sign (sharp-sign reader macro) 2-39
 - let** 5-36
 - let*** 5-36
 - lexical binding* 26-34
 - lexical closure* 26-34
 - lexical environment* 3-2, 26-34
 - lexical scope* 26-34
 - lexical variable* 26-34
 - linear-style conditional newline 22-17, 22-54
 - LISP package** A-1
 - Lisp image* 26-34
 - Lisp printer* 26-35
 - Lisp read-eval-print loop* 26-35
 - Lisp reader* 26-35
 - lisp-implementation-type** 25-28
 - lisp-implementation-version** 25-28
 - list* 14-2, 14-4, 26-35
 - list** 2-23, 2-26, 2-28, 7-26, 14-4, 14-20
 - list designator* 26-35
 - list structure* 26-35
 - list*** 14-20
 - list-all-packages** 11-14
 - list-length** 14-21
 - listen** 21-41
 - listp** 14-22
 - literal* 26-35
 - literal object* 3-14
 - load* 26-35
 - load** 3-60, 24-6
 - load time* 26-36

load time value 26–36
load-logical-pathname-translations 19–22
load-pathname 24–13
load-print 24–14
load-time-value 3–18, 3–63
:load-toplevel 3–60
load-truename 24–13
load-verbose 24–14
loader 26–36
:local 19–4
local declaration 3–29, 26–36
local precedence order 4–9, 4–12, 26–36
local slot 26–36
locally 3–100
log 12–45
logand 12–71
logandc1 12–71
logandc2 12–71
logbitp 12–73
logcount 12–74
logeqv 12–71
logical block 26–36
logical blocks 22–14
logical host 26–36
logical host designator 26–36
logical pathname 26–36
logical-pathname 19–15, 19–27
logical-pathname-translations 19–23
logior 12–71
lognand 12–71
lognor 12–71
lognot 12–71
logorc1 12–71
logorc2 12–71
logtest 12–75
logxor 12–71
long float 26–36
long-float 12–14
long-float-epsilon 12–88
long-float-negative-epsilon 12–88
long-site-name 25–28
loop 6–38
loop keyword 26–36
loop-finish 6–42
lower-case-p 13–20
lowercase 26–37
machine-instance 25–29
machine-type 25–30
machine-version 25–30
macro 3–18, 26–37
macro character 26–37
macro expansion 26–37
macro form 26–37
macro function 26–37
macro lambda list 3–41, 26–37
macro name 26–37
macro-function 3–73
macroexpand 3–75
macroexpand hook 26–37
macroexpand-1 3–75
macroexpand-hook 3–80
macrolet 3–18, 5–20
make-array 15–10
make-broadcast-stream 21–47
make-char A–1
make-concatenated-stream 21–51
make-condition 9–47
make-dispatch-macro-character 23–5
make-echo-stream 21–50
make-hash-table 18–4
make-instance 7–49
make-instances-obsolete 7–50
make-list 14–23
make-load-form 7–51
make-load-form-saving-slots 7–55
make-method 7–73
make-package 11–20
make-pathname 19–17
make-random-state 12–51
make-sequence 17–8
make-string 16–13
make-string-input-stream 21–53
make-string-output-stream 21–53
make-symbol 10–6
make-synonym-stream 21–46
make-two-way-stream 21–48
makunbound 10–21
mandatory-style conditional newline 22–54
map 17–10
map-into 17–12

-
- mapc** 14-40
 - mapcan** 14-40
 - mapcar** 14-40
 - mapcon** 14-40
 - maphash** 18-12
 - mapl** 14-40
 - maplist** 14-40
 - mapping* 26-37
 - mask-field** 12-81
 - max** 7-26, 12-22
 - member** 4-27, 14-39
 - member-if** 14-39
 - member-if-not** 14-39
 - merge** 17-30
 - merge-pathnames** 19-38
 - metaclass* 4-1, 4-9, 26-37
 - Metaobject Protocol* 26-37
 - method* 26-38
 - method** 4-24
 - method combination* 26-38
 - method-combination** 4-26, 25-18
 - method-combination-error** 9-26
 - method-defining form* 26-38
 - method-defining operator* 7-18, 26-38
 - method-qualifiers** 7-47
 - might signal 1-17
 - min** 7-26, 12-22
 - minimal compilation* 3-14, 26-38
 - Minus (sharpsign reader macro) 2-37
 - minusp** 12-24
 - miser-style conditional newline 22-17, 22-54
 - mismatch** 17-24
 - mod** 12-19, 12-47
 - modified lambda list* 26-38
 - *modules*** 24-14
 - most recent* 26-38
 - most-negative-double-float** 12-87
 - most-negative-fixnum** 12-82
 - most-negative-long-float** 12-87
 - most-negative-short-float** 12-87
 - most-negative-single-float** 12-87
 - most-positive-double-float** 12-87
 - most-positive-fixnum** 12-82
 - most-positive-long-float** 12-87
 - most-positive-short-float** 12-87
 - most-positive-single-float** 12-87
 - muffle-warning** 9-66, 9-68
 - multiple escape* 2-9, 26-38
 - multiple values* 26-38
 - multiple-value-bind** 5-79
 - multiple-value-call** 5-81
 - multiple-value-list** 5-81
 - multiple-value-prog1** 5-82
 - multiple-value-setq** 5-83
 - multiple-values-limit** 5-86
 - must signal 1-16
 - name* 1-7, 26-39
 - name-char** 13-25
 - named constant* 26-39
 - namespace* 3-1, 26-39
 - namestring* 19-1, 26-39
 - namestring** 19-28
 - nbutlast** 14-33
 - nconc** 7-26, 14-29
 - newline* 26-39
 - Newline (format directive) 22-42
 - next method* 7-23, 26-39
 - next-method-p** 7-72
 - nickname* 26-39
 - nil* 1-13, 26-39
 - nil** 1-13, 4-19, 5-54
 - nintersection** 14-52
 - ninth** 14-25
 - no-applicable-method** 7-47
 - no-next-method** 7-48
 - non-atomic* 26-39
 - non-constant variable* 26-39
 - non-correctable* 26-40
 - non-empty* 26-40
 - non-generic function* 26-40
 - non-graphic* 13-3, 26-40
 - non-list* 26-40
 - non-local exit* 26-40
 - non-nil* 26-40
 - non-null lexical environment* 26-40
 - non-simple* 26-40
 - non-terminating* 2-8, 26-40
 - non-top-level form* 26-40
 - normal return* 26-40
 - normalized* 26-40

not 4-27, 5-55
notany 5-67
notation 1-7
notevery 5-67
notinline 3-29, 3-93
nreconc 14-31
nreverse 17-17
nset-difference 14-56
nset-exclusive-or 14-58
nstring-capitalize 16-7
nstring-downcase 16-7
nstring-upcase 16-7
nsublis 14-13
nsubst 14-15
nsubst-if 14-15
nsubst-if-not 14-15
nsubstitute 17-26
nsubstitute-if 17-26
nsubstitute-if-not 17-26
nth 14-27
nth-value 5-86
nthcdr 14-37
null 13-2, 26-40
null 14-4, 14-28
null lexical environment 3-3, 26-41
number 26-41
number 12-11
numberp 12-54
numerator 12-61
numeric 26-41
nunion 14-61
O (format directive) 22-27
O (sharpsign reader macro) 2-33
object 26-41
object-traversing 26-41
oddp 12-40
open 21-2, 26-41
open 21-32
open-stream-p 21-12
operator 26-41
optimize 3-97
optimize quality 26-41
optional parameter 26-42
or 4-28, 5-71, 7-26
order of evaluation 3-63, 5-2, 5-43, 5-83, 6-5, 6-9, 7-4, 9-16, 12-79
ordinary function 26-42
ordinary lambda list 3-33, 26-42
otherwise 5-74, 5-77
otherwise inaccessible part 26-42
output 21-1, 26-42
output-stream-p 21-11
P (format directive) 22-40
P (sharpsign reader macro) 2-36
package 11-1, 26-42
package 11-8
package cell 26-42
package designator 26-42
package marker 26-42
package prefix 26-42
package registry 26-42
package 11-40
package-error 11-41
package-error-package 11-42
package-name 11-36
package-nicknames 11-37
package-shadowing-symbols 11-37
package-use-list 11-38
package-used-by-list 11-39
packagep 11-40
pairlis 14-46
pairwise 26-43
parallel 26-43
parameter 26-43
parameter specializer 26-43
parameter specializer name 26-43
parse-error 9-14
parse-integer 12-66
parse-namestring 19-30
pathname 19-1, 26-43
pathname 2-36, 19-15
pathname designator 26-43
pathname-device 19-20
pathname-directory 19-20
pathname-host 19-20
pathname-match-p 19-34
pathname-name 19-20
pathname-type 19-20
pathname-version 19-20
pathnamep 19-19

peek-char 21-17
 Percent (format directive) 22-25
phase 12-58
physical pathname 26-43
pi 12-32
place 5-1, 26-43
plist 26-43
 Plus (sharpsign reader macro) 2-37
plusp 12-24
pop 14-24
portable 26-43
position 17-21
position-if 17-21
position-if-not 17-21
potential copy 26-44
potential number 26-44
pprint 22-61
pprint dispatch table 22-16, 26-44
pprint-dispatch 22-47
pprint-exit-if-list-exhausted 22-48
pprint-fill 22-49
pprint-indent 22-50
pprint-linear 22-49
pprint-logical-block 22-51
pprint-newline 22-53
pprint-pop 22-55
pprint-tab 22-57
pprint-tabular 22-49
predicate 26-44
 prepared to signal 1-16
present 11-2, 24-1, 26-44
pretty print 26-44
pretty printer 22-14, 26-44
pretty printing stream 26-44
primary method 26-44
primary value 26-44
prin1 22-61
prin1-to-string 22-64
princ 22-61
princ-to-string 22-64
principal 26-44
print 22-61
print name 26-45
print-array 22-66
print-base 22-67
print-case 22-68
print-circle 2-36, 2-37, 22-69
print-escape 22-70
print-gensym 22-71
print-length 22-72
print-level 22-72
print-lines 22-73
print-miser-width 22-74
print-not-readable 22-79
print-not-readable-object 22-79
print-object 22-57
print-pprint-dispatch 22-75
print-pretty 22-75
print-radix 22-67
print-readably 22-76
print-right-margin 22-78
print-unreadable-object 22-59
printer control variable 22-1, 26-45
printer escaping 22-2, 26-45
printing 26-45
probe-file 20-3
process 3-14, 26-45
processor 26-45
proclaim 26-45
proclaim 3-81
proclamation 3-29, 26-45
prog 5-87
prog tag 26-45
prog* 5-87
prog1 5-89
prog2 5-89
progn 5-91, 7-26
program 26-45
program-error 5-104
programmer 26-45
programmer code 26-45
progv 5-38
proper list 14-2, 14-4, 26-45
proper name 26-45
proper sequence 26-46
proper subtype 26-46
property 26-46
property indicator 26-46
property list 26-46
property value 26-46

- provide** 24–15
- psetf** 5–99
- psetq** 5–40
- purports to conform* 26–46
- push** 14–23
- pushnew** 14–55
- qualified method* 26–46
- qualifier* 26–46
- query I/O* 26–46
- *query-io*** 21–57
- Question-Mark (format directive) 22–39
- quotation (of forms) 2–24, 2–26, 2–28
- quotation (of strings) 2–26
- quote** 2–24, 2–26, 2–28, 3–65
- quoted object* 26–46
- quux** 1–15
- R (format directive) 22–25
- R (sharpsign reader macro) 2–34
- radix* 26–47
- random** 12–52
- random state* 26–47
- random-state** 12–50
- *random-state*** 12–53
- random-state-p** 12–53
- rank* 15–1, 26–47
- rassoc** 14–47
- rassoc-if** 14–47
- rassoc-if-not** 14–47
- ratio* 26–47
- ratio** 12–16, 22–3
- ratio marker* 26–47
- rational* 26–47
- rational** 12–16, 12–62
- rationalize** 12–62
- rationalp** 12–63
- read* 26–47
- read** 23–6
- *read-base*** 2–33, 2–34, 23–19
- read-byte** 21–15
- read-char** 21–18
- read-char-no-hang** 21–19
- *read-default-float-format*** 23–19
- read-delimited-list** 23–8
- *read-eval*** 2–33, 23–20
- read-from-string** 23–10
- read-line** 21–24
- read-preserving-whitespace** 23–6
- read-sequence** 21–26
- *read-suppress*** 23–21
- readably* 26–47
- reader* 26–47
- reader macro* 26–47
- reader macro function* 2–8, 26–47
- reader-error** 23–23
- readtable* 2–1, 26–47
- readtable** 23–4
- readtable case* 26–47
- readtable designator* 26–48
- *readtable*** 23–22
- readtable-case** 23–11
- readtablep** 23–12
- real** 12–12
- realp** 12–60
- realpart** 12–59
- recognizable subtype* 26–48
- redefinition 11–5
- reduce** 17–13
- reference* 26–48
- registered package* 26–48
- reinitialize-instance** 7–31
- relative* 26–48
- :relative** 19–7
- rem** 12–47
- remf** 14–51
- remhash** 18–11
- remove** 17–32
- remove-duplicates** 17–35
- remove-if** 17–32
- remove-if-not** 17–32
- remove-method** 7–49
- remprop** 10–19
- rename-file** 20–8
- rename-package** 11–14
- repertoire* 13–2, 26–48
- replace** 17–25
- report* 26–48
- report message* 9–4, 26–48
- require** 24–15
- required parameter* 26–48
- rest** 14–38

- rest list* 26-48
- rest parameter* 26-48
- restart* 26-48
- restart** 9-48
- restart designator* 26-49
- restart function* 26-49
- restart-bind** 9-54
- restart-case** 9-56
- restart-name** 9-61
- return* 26-49
- return** 5-47
- return value* 26-49
- return-from** 5-45
- revappend** 14-31
- reverse** 17-17
- Right-Brace (format directive) 22-39
- Right-Bracket (format directive) 22-37
- Right-Paren (format directive) 22-40
- right-parenthesis* 26-49
- Right-Parenthesis (reader macro) 2-23
- room** 25-21
- rotatef** 5-103
- round** 12-25
- row-major-aref** 15-28
- rplaca** 14-8
- rplacd** 14-8
- run time* 3-15, 26-49
- run-time compiler* 3-15, 26-49
- run-time definition* 3-15, 26-49
- run-time environment* 3-15, 26-49
- S (format directive) 22-31
- S (sharpsign reader macro) 2-36
- safe* 1-15, 26-49
- safe call* 3-50, 26-49
- safety** 3-29, 3-97
- same* 26-49
- satisfies** 4-26
- satisfy the test* 17-2, 17-4, 26-50
- sbit** 15-36
- scale-float** 12-82
- schar** 16-5
- scope* 26-50
- script* 26-50
- search** 17-23
- second** 14-25
- secondary value* 26-50
- section* 26-50
- sections* 22-14
- self-evaluating object* 26-50
- semi-standard* 26-50
- semicolon* 26-51
- Semicolon (format directive) 22-40
- Semicolon (reader macro) 2-24
- sequence* 17-1, 26-51
- sequence** 17-5
- sequence function* 17-1, 26-51
- sequential* 26-51
- sequentially* 26-51
- serious condition* 26-51
- serious-condition** 9-12
- session* 26-51
- set* 26-51
- set** 10-22
- set-char-bit** A-1
- set-difference** 14-56
- set-dispatch-macro-character** 23-13
- set-exclusive-or** 14-58
- set-macro-character** 23-14
- set-pprint-dispatch** 22-60
- set-syntax-from-char** 23-16
- setf** 5-99, 25-18
- setf expander* 26-51
- setf expansion* 26-51
- setf function* 26-51
- setf function name* 26-51
- setq** 5-39
- seventh** 14-25
- shadow* 3-11, 4-8, 26-51
- shadow** 11-15
- shadowing symbol* 11-3, 11-4, 26-51
- shadowing symbols list* 26-51
- shadowing-import** 11-16
- shared slot* 26-52
- shared-initialize** 7-32
- sharpsign* 26-52
- Sharpsign (reader macro) 2-29
- Sharpsign (sharpsign reader macro) 2-37
- Sharpsign A (reader macro) 2-35
- Sharpsign Asterisk (reader macro) 2-32
- Sharpsign B (reader macro) 2-33

- Sharpsign Backslash (reader macro) 2-31
- Sharpsign C (reader macro) 2-34
- Sharpsign Colon (reader macro) 2-33
- Sharpsign Dot (reader macro) 2-33
- Sharpsign Equal-Sign (reader macro) 2-36
- Sharpsign Left-Parenthesis (reader macro) 2-31
- Sharpsign Less-Than-Sign (reader macro) 2-39
- Sharpsign Minus (reader macro) 2-37
- Sharpsign O (reader macro) 2-33
- Sharpsign P (reader macro) 2-36
- Sharpsign Plus (reader macro) 2-37
- Sharpsign R (reader macro) 2-34
- Sharpsign Right-Parenthesis 2-40
- Sharpsign S (reader macro) 2-36
- Sharpsign Sharpsign (reader macro) 2-37, 22-56

- Sharpsign Single-Quote (reader macro) 2-31
- Sharpsign Vertical-Bar (reader macro) 2-38
- Sharpsign Whitespace 2-39, 2-40
- Sharpsign X (reader macro) 2-34
- shiftf** 5-101
- short float* 26-52
- short-float** 12-14
- short-float-epsilon** 12-88
- short-float-negative-epsilon** 12-88
- short-site-name** 25-28
- should signal 1-16
- sign* 26-52
- signal* 1-16, 1-17, 26-52
- signal** 9-27
- signature* 26-52
- signed-byte** 12-17
- signum** 12-48
- similar* 3-23, 26-52
- similarity* 26-52
- simple* 26-52
- simple array* 26-52
- simple bit array* 26-52
- simple bit vector* 26-52
- simple condition* 26-52
- simple general vector* 26-52
- simple string* 26-52
- simple vector* 26-53
- simple-array** 15-6
- simple-base-string** 16-4
- simple-bit-vector** 2-32, 15-10
- simple-bit-vector-p** 15-40
- simple-condition** 9-28
- simple-condition-format-arguments** 9-29
- simple-condition-format-control** 9-29
- simple-error** 9-25
- simple-string** 16-3
- simple-string-p** 16-4
- simple-type-error** 4-42
- simple-vector** 2-31, 15-8
- simple-vector-p** 15-31
- simple-warning** 9-31
- sin** 12-28
- single escape* 2-9, 26-53
- single float* 26-53
- single-float** 12-14
- single-float-epsilon** 12-88
- single-float-negative-epsilon** 12-88
- single-quote* 26-53
- Single-Quote (reader macro) 2-24
- Single-Quote (sharp-sign reader macro) 2-31
- singleton* 26-53
- sinh** 12-33
- situation* 26-53
- sixth** 14-25
- slash* 26-53
- Slash (format directive) 22-33
- sleep** 25-7
- slot* 26-53
- slot specifier* 4-10, 26-53
- slot-boundp** 7-40
- slot-exists-p** 7-41
- slot-makunbound** 7-42
- slot-missing** 7-43
- slot-unbound** 7-44
- slot-value** 7-45
- software-type** 25-31
- software-version** 25-31
- some** 5-67
- sort** 17-18
- source code* 24-1, 26-53
- source file* 24-1, 26-53
- space* 26-53
- space** 3-97
- special** 3-29, 3-98

- special form* 26-53
- special operator* 26-53
- special variable* 26-53
- special-operator-p** 3-103
- specialize* 26-54
- specialized* 26-54
- specialized lambda list* 3-40, 26-54
- speed** 3-97
- spreadable argument list designator* 26-54
- sqrt** 12-49
- stable-sort** 17-18
- stack allocate* 26-54
- stack-allocated* 26-54
- standard** 7-24, 7-26
- standard character* 2-2, 26-54
- standard class* 26-54
- standard generic function* 26-54
- standard input* 26-54
- standard method combination* 26-54
- standard object* 26-54
- standard output* 26-55
- standard pprint dispatch table* 26-55
- standard readtable* 2-1, 26-55
- standard syntax* 2-1, 26-55
- standard-char** 13-9
- standard-char-p** 13-18
- standard-class** 4-24
- standard-generic-function** 4-23
- *standard-input*** 21-57
- standard-method** 4-25
- standard-object** 4-25
- *standard-output*** 21-57
- standardized* 26-55
- startup environment* 3-14, 26-55
- step* 26-55
- step** 25-13
- storage-condition** 9-15
- store-value** 9-67, 9-68
- stream* 20-1, 21-1, 21-2, 26-55
- stream** 21-7
- stream associated with a file* 20-1, 26-55
- stream designator* 26-56
- stream element type* 26-56
- stream variable* 21-4, 26-56
- stream variable designator* 26-56
- stream-element-type** 21-13
- stream-error** 21-60
- stream-error-stream** 21-60
- stream-external-format** 21-36
- streamamp** 21-14
- string* 15-3, 26-56
- string** 2-26, 16-2, 16-6
- string designator* 26-56
- string equal* 26-56
- string stream* 26-56
- string-capitalize** 16-7
- string-char** A-1
- string-char-p** A-1
- string-downcase** 16-7
- string-equal** 16-10
- string-greaterp** 16-10
- string-left-trim** 16-9
- string-lessp** 16-10
- string-not-equal** 16-10
- string-not-greaterp** 16-10
- string-not-lessp** 16-10
- string-right-trim** 16-9
- string-stream** 21-10
- string-trim** 16-9
- string-upcase** 16-7
- string/=** 16-10
- string<** 16-10
- string<=** 16-10
- string=** 16-10
- string>** 16-10
- string>=** 16-10
- stringp** 16-12
- structure* 26-56
- structure** 2-36, 25-19
- structure class* 26-56
- structure name* 26-56
- structure-class** 4-24
- structure-object** 4-25
- style warning* 26-56
- style-warning** 9-12
- subclass* 4-8, 26-56
- subexpression* 26-57
- subform* 26-57
- sublis** 14-13
- subrepertoire* 26-57

-
- subseq** 17-9
 - subsetp** 14-59
 - subst** 14-15
 - subst-if** 14-15
 - subst-if-not** 14-15
 - substitute** 17-26
 - substitute-if** 17-26
 - substitute-if-not** 17-26
 - subtype* 26-57
 - subtypep** 4-34
 - superclass* 4-8, 26-57
 - supertype* 26-57
 - supplied-p parameter* 26-57
 - svref** 15-32
 - sxhash** 18-15
 - symbol* 26-57
 - symbol** 2-33, 10-2
 - symbol macro* 3-18, 26-57
 - symbol-function** 10-11
 - symbol-macrolet** 3-18, 3-79
 - symbol-name** 10-13
 - symbol-package** 10-14
 - symbol-plist** 10-15
 - symbol-value** 10-16
 - symbolp** 10-4
 - synonym stream* 26-57
 - synonym stream symbol* 26-57
 - synonym-stream** 21-10
 - synonym-stream-symbol** 21-46
 - syntax type* 2-5, 26-57
 - SYSTEM package** A-1
 - system class* 26-58
 - system code* 26-58
 - t* 26-58
 - t** 4-26, 5-55, 5-74, 5-77, 25-18, 25-19
 - T (format directive)** 22-34
 - tag* 26-58
 - tagbody** 5-48
 - tail* 26-58
 - tailp** 14-35
 - tan** 12-28
 - tanh** 12-33
 - target* 26-58
 - tenth** 14-25
 - terminal I/O* 26-58
 - *terminal-io*** 21-59
 - terminating* 2-8, 26-58
 - terpri** 21-21
 - tertiary value* 26-58
 - the** 3-102
 - third** 14-25
 - throw* 26-58
 - throw** 5-49
 - tilde* 26-58
 - Tilde (format directive) 22-25
 - Tilde A (format directive) 22-31
 - Tilde Ampersand (format directive) 22-25
 - Tilde Asterisk (format directive) 22-36
 - Tilde B (format directive) 22-26
 - Tilde C (format directive) 22-24
 - Tilde Circumflex (format directive) 22-41
 - Tilde D (format directive) 22-26
 - Tilde Dollarsign (format directive) 22-30
 - Tilde E (format directive) 22-28
 - Tilde F (format directive) 22-27
 - Tilde G (format directive) 22-30
 - Tilde Greater-Than-Sign (format directive) 22-36
 - Tilde I (format directive) 22-33
 - Tilde Left-Brace (format directive) 22-37
 - Tilde Left-Bracket (format directive) 22-36
 - Tilde Left-Paren (format directive) 22-39
 - Tilde Less-Than-Sign (format directive) 22-32, 22-35
 - Tilde Newline (format directive) 22-42
 - Tilde O (format directive) 22-27
 - Tilde P (format directive) 22-40
 - Tilde Percent (format directive) 22-25
 - Tilde Question-Mark (format directive) 22-39
 - Tilde R (format directive) 22-25
 - Tilde Right-Brace (format directive) 22-39
 - Tilde Right-Bracket (format directive) 22-37
 - Tilde Right-Paren (format directive) 22-40
 - Tilde S (format directive) 22-31
 - Tilde Semicolon (format directive) 22-40
 - Tilde Slash (format directive) 22-33
 - Tilde T (format directive) 22-34
 - Tilde Tilde (format directive) 22-25
 - Tilde Underscore (format directive) 22-32
 - Tilde Vertical-Bar (format directive) 22-25

Tilde W (format directive) 22-32
Tilde X (format directive) 22-27
time 26-58
time 25-14
time zone 26-59
token 2-6, 26-59
top level form 26-59
trace 25-12
trace output 26-59
trace-output 21-57
translate-logical-pathname 19-35
translate-pathname 19-36
tree 14-1, 26-59
tree structure 26-59
tree-equal 14-18
true 26-59
truename 20-2, 26-59
truename 20-5
truncate 12-25
two-way stream 26-59
two-way-stream 21-11
two-way-stream-input-stream 21-49
two-way-stream-output-stream 21-49
type 26-59
type 3-90, 25-19
type declaration 26-60
type equivalent 26-60
type expand 26-60
type specifier 26-60
type-error 4-41
type-error-datum 4-41
type-error-expected-type 4-41
type-of 4-37
typecase 5-76
typep 4-39
unbound 26-60
unbound variable 26-60
unbound-slot 7-90
unbound-slot-instance 7-91
unbound-variable 10-23
undefined consequences 1-17
undefined function 26-60
undefined-function 5-104
Underscore (format directive) 22-32
unexport 11-24
unintern 26-60
unintern 11-25
uninterned 26-60
union 14-61
universal time 25-3, 26-60
unless 5-72
unqualified method 26-60
unread-char 21-22
unregistered package 26-60
unsafe 1-16, 26-60
unsafe call 3-51, 26-61
unsigned-byte 12-18
:unspecific 19-6
unspecified consequences 1-17
unspecified values 1-17
untrace 25-12
unuse-package 11-27
unwind-protect 5-51
:up 19-8
update-instance-for-different-class 7-34
update-instance-for-redefined-class 7-35
upgrade 26-61
upgraded array element type 15-3, 26-61
upgraded complex part type 26-61
upgraded-array-element-type 15-29
upgraded-complex-part-type 12-60
upper-case-p 13-20
uppercase 26-61
use 26-61
use list 26-61
use-package 11-28
use-value 9-68
user 26-61
USER package A-1
user-homedir-pathname 25-32
valid array dimension 26-61
valid array index 26-62
valid array row-major index 26-62
valid fill pointer 26-62
valid logical pathname host 26-62
valid pathname device 26-62
valid pathname directory 26-62
valid pathname host 26-62
valid pathname name 26-62
valid pathname type 26-62

valid pathname version 26-62
valid physical pathname host 26-62
valid sequence index 26-62
value 26-63
value cell 26-63
values 4-29, 5-84
values-list 5-85
variable 26-63
variable 25-19
vector 15-1, 26-63
vector 2-31, 15-7, 15-33
vector-pop 15-33
vector-push 15-34
vector-push-extend 15-34
vectorp 15-36
vertical-bar 26-63
Vertical-Bar (format directive) 22-25
Vertical-Bar (sharpsign reader macro) 2-38
W (format directive) 22-32
warn 9-29
warning 9-11
warning 1-18
when 5-72
whitespace 26-63
wild 26-63
:wild 19-5, 19-8
:wild-inferiors 19-5, 19-8
wild-pathname-p 19-33
with-accessors 7-56
with-compilation-unit 24-8
with-condition-restarts 9-62
with-hash-table-iterator 18-13
with-input-from-string 21-54
with-open-file 21-37
with-open-stream 21-40
with-output-to-string 21-55
with-package-iterator 11-21
with-simple-restart 9-63
with-slots 7-58
with-standard-io-syntax 23-17
write 26-63
write 22-61
write-byte 21-16
write-char 21-23
write-line 21-25
write-sequence 21-27
write-string 21-25
write-to-string 22-64
writer 26-63
X (format directive) 22-27
X (sharpsign reader macro) 2-34
y-or-n-p 21-44
yes-or-no-p 21-44
yield 26-64
zerop 12-25
' 2-26

Credits

Principal Technical Editors:

Kent M. Pitman	Harlequin, Inc.	1993-present
	Symbolics, Inc.	1990-1992
Kathy Chapman	Digital Equipment Corporation	1987-1989

Occasional Guest Editors:

Richard P. Gabriel	Lucid, Inc.
Sandra Loosemore	self

Financial Contributors to the Editing Process:

Digital Equipment Corporation
Harlequin, Ltd. and Harlequin, Inc.
Symbolics, Inc.
Apple, Inc.
Franz, Inc.
Lucid, Inc.

Special thanks to Guy L. Steele Jr. and Digital Press for producing *Common Lisp: The Language*, and for relaxing copyright restrictions enough to make it possible for that document's text to provide an early basis of this work.

Edit and Review History:

01-Jan-89	Chapman	Draft of Chapters 1.1 (scope).
01-Jan-89	Pitman	Draft of Chapters 5.1 (conditions).
01-May-89	Chapman	Draft of 1.2–1.6.
01-May-89	Gabriel	Rewrite of Chapters 1.1 and 5.1.
01-Jun-89	Loosemore	Review of Chapter 4.2.
01-Jun-89	Pitman	Review of Glossary
15-Jun-89	Gabriel	Rewrite of Glossary
16-Jun-89	Margolin	Comments on Chapters 2.1–2.4 (types, objects).
23-Jun-89	Gabriel	Rewrite of 4.2.
07-Jul-89	Moon	Review of Chapters 4.1, 4.3
12-Jul-89	Gabriel	Revision of 4.2.
15-Jul-89	Pitman	Review of Glossary
18-Jul-89	Gray	Comments on 5.1
25-Jul-89	Gabriel	Revision of Chapters 1.2–1.6, 2.2
26-Jul-89	Gabriel	Rewrite of 5.1
26-Jul-89	Gabriel	Rewrite of 4.1.
27-Jul-89	Pitman	Revision of 5.1
27-Jul-89	Gabriel	Revision of 5.1
28-Jul-89	Chapman	Draft of 2.2, 3.2, 3.3, 5.4
28-Jul-89	Gabriel	Revision of Glossary.
01-Oct-89	Margolin	Review of Dictionary from Jun-89 draft.
20-Jan-91	Pitman	Draft 8.81 (for X3J13 review). Document X3J13/91-101.
29-Jan-91	Waters	Review of 8.81/Chapter 23 (Printer).
01-Mar-91	Moon	Review of 8.81/Chapter 4 (Evaluation and Compilation).
01-Mar-91	Barrett	Review of 8.81/Chapter 4 (Evaluation and Compilation).
01-Mar-91	Moon	Review of 8.81/Glossary.
13-Mar-90	Wechsler	Review of 8.81/Glossary.
21-Mar-91	Kerns	Review of 8.81/Chapter 1.
26-Apr-91	Margolin	Review of 8.81/Chapters 1–12.
15-May-91	Barrett	Review of 8.81/Chapters 5 (Misc), 11 (Conditions).
04-Jun-91	Laddaga	Review of 9.60/Chapter 20 (Pathnames).
10-Jun-91	Pitman	Draft 9.126 (for X3J13 review). Document X3J13/91-102.
02-Sep-91	Barrett	Review of 9.28/Chapter 4 (Evaluation and Compilation).
02-Sep-91	Barrett	Review of 9.52/Chapter 4 (Evaluation and Compilation).
15-Sep-91	Barrett	Review of 9.126/Chapter 4 (Evaluation and Compilation) and Chapter 7 (Evaluation/Compilation). (some comments not yet merged)
18-Sep-91	Wechsler	Review of 9.126.
21-Sep-91	Barrett	Review of 10.16/Chapter 7 (Evaluation/Compilation). (some comments not yet merged)
28-Sep-91	Barrett	Review of 10.95/Chapter 25 (Printer). (some comments not yet merged)

13-Oct-91	Barrett	Review (and help editing) of 10.104/Chapter 4 (Evaluation and Compilation)
15-Oct-91	Waters	Review of 10.95/Chapter 25 (Printer).
24-Oct-91	Pitman	Draft 10.156 (for X3J13 review). Document X3J13/91-103.
04-Nov-91	Moon	Review of 10.156/Chapter 5 (Data and Control Flow) and Chapter 26 (Glossary).
11-Nov-91	Loosemore	Review of 10.156/Chapter 2 (Syntax), Chapter 3 (Evaluation and Compilation), Chapter 5 (Data and Control Flow), and Chapter 8 (Structures).
02-Dec-91	Barrett	Review of 10.156/Chapter 4 (Types and Classes), and Chapter 10 (Symbols).
02-Dec-91	Barrett	Review of 10.156/Chapter 3 (Evaluation and Compilation), Chapter 6 (Iteration), Chapter 9 (Conditions), and Chapter 14 (Conses). (some comments not yet merged)
09-Dec-91	Gabriel	Review of 10.156/Chapter 1 (Introduction), Chapter 2 (Syntax), and Chapter 3 (Evaluation and Compilation).
09-Dec-91	Ida	Light review of 10.156/Chapters 1-5.
09-Dec-91	Moon	Review of 10.156/Chapter 3 (Evaluation and Compilation). (some comments not yet merged)
10-Dec-91	Loosemore	Review of 10.156/Chapter 10 (Symbols), Chapter 20 (Files), and Chapter 13 (Characters).
10-Dec-91	Loosemore	Review of 10.156/Chapter 14 (Conses). (some comments not yet merged)
10-Dec-91	Laubsch	Review of 10.156/Chapters 1 (Introduction), Chapter 2 (Syntax), Chapter 3 (Evaluation and Compilation), Chapter 4 (Types and Classes), Chapter 5 (Data and Control Flow), Chapter 7 (Objects), Chapter 11 (Packages), Chapter 19 (Filenames), and Chapter 21 (Streams).
18-Dec-91	Margolin	Review of 10.156/Chapter 18 (Hash Tables).
04-Jan-92	White	Review of 10.156/Chapter 6 (Iteration), Chapter 11 (Packages), Chapter 18 (Hash Tables), and Chapter 23 (Reader).
04-Jan-92	White	Review of 10.156/Chapter 26 (Glossary). (some comments not yet merged)
04-Jan-92	Barrett	Review of 10.156/Chapter 18 (Hash Tables) and Chapter 16 (Strings).
04-Jan-92	Barrett	Review of 10.156/Chapter 15 (Arrays) and Chapter 21 (Streams). (some comments not yet merged)
06-Jan-92	Loosemore	Review of 10.156/Chapter 16 (Strings), Chapter 17 (Sequences), and Chapter 25 (Environment).
06-Jan-92	Loosemore	Review of 10.156/Chapter 21 (Streams) and Chapter 23 (Reader). (some comments not yet merged)
06-Jan-92	Margolin	Review of 10.156/Chapter 2 (Syntax).
07-Jan-92	Margolin	Review of 10.156/Chapter 4 (Types and Classes).
03-Feb-92	Aspinall	Review of 10.156/Chapter 12 (Numbers).

16-Feb-92	Pitman	Draft 11.82 (for X3J13 letter ballot). Document X3J13/92-101.
16-Mar-92	Loosemore	Review of 11.82/Chapter 1, 3, 4, 5, 7, 8, 9, 10, 11, 12, 18, 22, 23, 24, 25, and 26.
16-Feb-92	Pitman	Draft 12.24 (for X3 consideration). Document X3J13/92-102.
09-Sep-92	Samson	Public Review Comments (#1). Documents X3J13/92-1001 to 92-1003.
22-Oct-92	Rose, Yen	Public Review Comments (#2). Documents X3J13/92-1101 to 92-1103.
23-Oct-92	Staley	Public Review Comments (#3). Documents X3J13/92-1201 to 92-1204.
09-Nov-92	Barrett	Public Review Comments (#4). Documents X3J13/92-3101 to 92-3110.
11-Nov-92	Moon	Public Review Comments (#5). Documents X3J13/92-3201 to 92-3248.
17-Nov-92	Loosemore	Public Review Comments (#6). Documents X3J13/92-1301 to 92-1335.
23-Nov-92	Margolin	Public Review Comments (#7). Documents X3J13/92-1401 to 92-1419.
23-Nov-92	Withington	Public Review Comments (#8a). Documents X3J13/92-1501 to 92-1512.
23-Nov-92	Feinberg	Public Review Comments (#8b). Documents X3J13/92-1601 to 92-1603.
23-Nov-92	Wechsler	Public Review Comments (#8c). Documents X3J13/92-1701 to 92-1703.
23-Nov-92	Moore	Public Review Comments (#9). Documents X3J13/92-1801 to 92-1802.
23-Nov-92	Flanagan	Public Review Comments (#10). Documents X3J13/92-1901 to 92-1910.
23-Nov-92	Dalton	Public Review Comments (#11). Documents X3J13/92-2001 to 92-2012.
23-Nov-92	Gallagher	Public Review Comments (#12). Documents X3J13/92-2101 to 92-2103.
23-Nov-92	Norvig	Public Review Comments (#13). Documents X3J13/92-2201 to 92-2208.
24-Nov-92	Robertson	Public Review Comments (#14). Document X3J13/92-2301.
23-Nov-92	Kawabe	Public Review Comments (#15). Documents X3J13/92-2401 to 92-2403.
23-Nov-92	Barrett	Public Review Comments (#16). Documents X3J13/92-2511 to X3J13/92-2531.
23-Nov-92	Wertheimer	Public Review Comments (#17). Document X3J13/92-2601.
24-Nov-92	Pitman	Public Review Comments (#18). Documents X3J13/92-2701 to 92-2742.
24-Nov-92	Mato Mira	Public Review Comments (#19). Documents X3J13/92-2801 to 92-2805.
24-Nov-92	Philpot	Public Review Comments (#20). Document X3J13/92-2901.
23-Nov-92	Cerys	Public Review Comments (#21). Document X3J13/92-3001.
30-Aug-93	Pitman	Draft 13.65 (for X3J13 consideration). Document X3J13/93-101.
04-Oct-93	X3J13	Minor fixes to Draft 13.65 before sending to X3.
05-Oct-93	Pitman	Draft 14.10 (for X3 consideration). Document X3J13/93-102.
08-Nov-93	Dalton	“reply to reply to pr comments”. Document X3J13/94-311.
04-Apr-94	Boyer, Kaufmann, Moore	Public Review Comments (#1). Document X3J13/94-305.
05-Apr-94	Pitman	Public Review Comments (#2). Document X3J13/94-306.
14-Mar-94	Schulenburg	Public Review Comments (#3). Document X3J13/94-307.
04-Apr-94	Shepard	Late commentary. Document X3J13/94-309.
05-May-94	X3J13	Editorial-only changes to Draft 14.10 in response to comments.
10-May-94	Pitman	Draft 15.17 (for X3 consideration). Document X3J13/94-101.
12-Aug-94	X3J13	Letter ballot to make specific corrections to Credits. Drafts 15.17 and 15.17R are <i>identical</i> except for: Changes to document date and version number. Disclaimer added to back of cover page. Changes to this Edit and Review History, page <i>Credits iv</i> . Changes to names and headings, pages <i>Credits v-vii</i> .
12-Aug-94	Pitman	Draft 15.17R (for X3 consideration). Document X3J13/94-101R.

The following lists of information are almost certainly incomplete, but it was felt that it was better to risk publishing incomplete information than to fail to acknowledge important contributions by the many people and organizations who have contributed to this effort.

Mention here of any individual or organization does not imply endorsement of this document by that individual or organization.

Ad Hoc Group Chairs:

Characters	Linden, Thom
Charter	Ennis, Susan P.
Compiler Specification	Haflich, Steven M.
	Loosemore, Sandra
Editorial	Chapman, Kathy
	van Roggen, Walter
Error and Condition System	Pitman, Kent M.
Graphics & Windows	Douglas Rand
	Schoen, Eric
Iteration Facility	White, JonL
Language Cleanup	Masinter, Larry
	Fahlman, Scott
Lisp ₁ /Lisp ₂	Gabriel, Richard P.
Macros	Haflich, Steven M.
	Pitman, Kent M.
	Wegman, Mark
Object System	Bobrow, Daniel G.
Presentation of Standard	Brown, Gary L.
Pretty Printer	Waters, Richard C.
Public Review	Ida, Masayuki
Types & Declarations	Scherlis, William L.
Validation	Berman, Richard

Major Administrative Contributions:

Barrett, Kim	Mathis, Robert
Brown, Gary L.	Pitman, Kent M.
Eiron, Hanoch	Steele, Guy L., Jr.
Gabriel, Richard P.	Tyson, Mabry
Haflich, Steven M.	Van Deusen, Mary
Ida, Masayuki	White, JonL
Loeffler, David D.	Whittemore, Susan
Loosemore, Sandra	Woodyatt, Anne
Masinter, Larry	Zubkoff, Jan L.

Major Technical Contributions:

Barrett, Kim A.	Keene, Sonya	Moon, David A.
Bobrow, Daniel G.	Kempf, James	Perdue, Crispin
Daniels, Andy	Kerns, Robert W.	Pitman, Kent M.
DeMichiel, Linda G.	Kiczales, Gregor	Steele, Guy L., Jr.
Dussud, Patrick H.	Loosemore, Sandra	Waters, Richard C.
Fahlman, Scott	Margolin, Barry	Weinreb, Daniel
Gabriel, Richard P.	Masinter, Larry	White, JonL
Ida, Masayuki	Mlynarik, Richard	

Participating Companies and Organizations:

AI Architects, Inc.	Lucid, Inc.
Amoco Production Co.	MCC
Aoyama Gakuin University	MIT
Apple Computer	MITRE Corporation
Boeing Advanced Technology Center	MSC
Carnegie-Mellon University	NASA Ames Research Center
Chestnut Software	National Bureau of Standards
Computer Sciences	Nihon Symbolics
Computer & Business Equipment Manufacturing Association (X3 Secretariat)	
CONTEL	ParcPlace Systems, Inc.
CSC	Prime Computer
DARPA	Siemens
Digital Equipment Corporation	Southern Illinois University
Encore	Sperry
Evans & Sutherland	SRI International
Franz, Inc.	Sun Microsystems
Gigamos	Symbolics
GMD	Tektronix
Gold Hill	Texas Instruments
Grumman Data Systems Corporation	The Aerospace Corporation
Harlequin, Ltd.	Thinking Machines Corporation
Hewlett-Packard	Unisys
Honeywell	University of Bath
IBM	University of Edinburgh
Ibuki	University of Maryland
Integrated Inference Machines	University of Utah
International LISP Associates	US Army
Johnson Controls, Inc.	USC/ISI
LMI	Xerox

Individual Participants:

Adler, Annette	Hafflich, Steven M.	Peck, Jeff
Allen, Stanley	Harris, Richard M.	Pellegrino, Bob
Antonisse, Jim	Hendler, Jim	Perdue, Crispin
Arbaugh, Bill	Hewitt, Carl	Philipp, Christopher
Balzer, Bob	Hornig, Charles	Pierson, Dan
Barrett, Kim	Ida, Masayuki	Pitman, Kent M.
Bartley, David H.	Kachurik, Catherine A.	Posner, Dave
Beckerle, Michael	Kahn, Ken	Raghavan, B.
Beiser, Paul	Keene, Sonya	Rand, Douglas
Benson, Eric	Keller, Shaun	Rininger, Jeff
Berman, Richard	Kempf, James	Rosenking, Jeffrey P.
Bobrow, Daniel G.	Kerns, Robert W.	Scherlis, William L.
Boelk, Mary P.	Kiczales, Gregor	Shiota, Eiji
Brittain, Skona	Kolb, Dieter	Sizer, Andy
Brown, Gary L.	Koschmann, Timothy	Slater, David
Chailloux, Jerome	Kosinski, Paul	Sodan, Angela
Chapman, Kathy	Larson, Aaron	Soley, Richard M.
Clinger, Will	Latto, Andy	Squires, Stephen L.
Coffee, Peter C.	Laubsch, Joachim	St. Clair, Bill
Cugini, John	Layer, Kevin	Stanhope, Philip
Curtis, Pavel	Linden, Thom	Steele, Guy L., Jr.
Dabrowski, Christopher	Loeffler, David D.	Tucker, Paul
Daessler, Klaus	Loosemore, Sandra	Turba, Thomas
Dalton, Jeff	Magataca, Mituhiko	Unietis, Dave
Daniels, Andy	Margolin, Barry	Van Deusen, Mary
DeMichiel, Linda G.	Masinter, Larry	van Roggen, Walter
Doi, Takumi	Mathis, Robert	Waldrum, Ellen
Drescher, Gary	Matthews, David C.	Waters, Richard C.
Duggan, Jerry	McCarthy, John	Wechsler, Allan
Dussud, Patrick H.	Mikelsons, Martin	Wegman, Mark
Ennis, Susan P.	Mlynarik, Richard	Weinreb, Daniel
Fahlman, Scott	Moon, David A.	Weyhrauch, Richard
Frayman, Felix	Moore, Timothy	White, JonL
Gabriel, Richard P.	Nicoud, Stephen	Wieland, Alexis
Giansiracusa, Bob	Nilsson, Jarl	Withington, P. Tucker
Goldstein, Brad	O'Dell, Jim	Wright, Whitman
Gray, David	Ohlander, Ron	York, Bill
Greenblatt, Richard	Padget, Julian	Zacharias, Gail
Hadden, George D.	Palter, Gary	Zubkoff, Jan L.

Version 15.17R, X3J13/94-101R.
Fri 12-Aug-1994 6:35pm EDT
