

A fast dynamic language for technical computing

Stefan Karpinski

Jeff Bezanson

Viral B Shah

Alan Edelman

Numerical languages



What's the deal with these?

specialized for numerical work

Matlab:

everything is a <u>complex matrix</u>

R (and S before it):

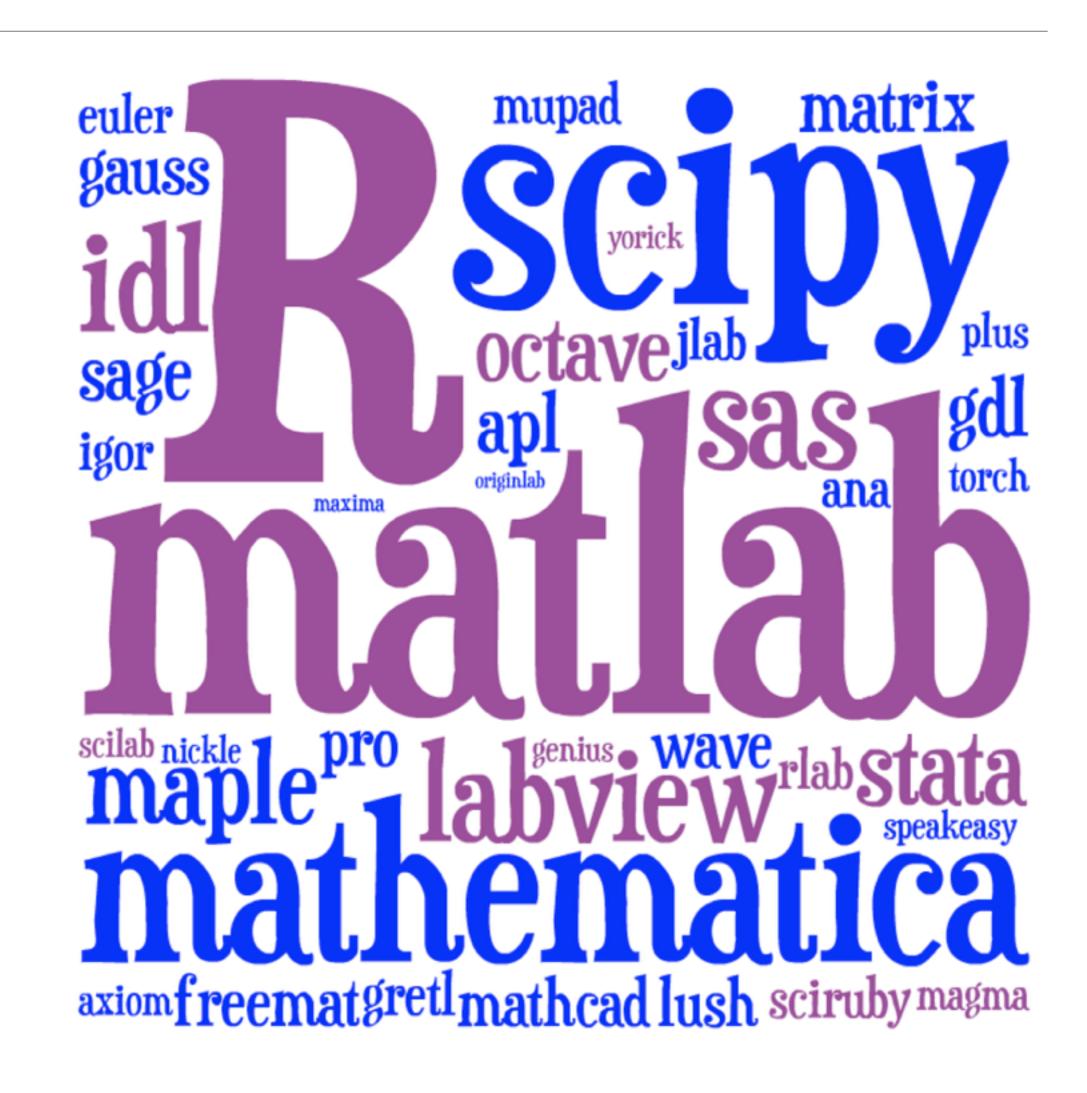
- ▶ allow "NA" values everywhere
- data frame as basic data type

Mathematica:

symbolic rewriting everywhere

NumPy:

typed arrays for Python



Are C and Scheme numerical?



Scheme R⁶RS spec:

▶ 20% numerical

C99 spec:

▶ 20% numerical

Introduction	CONTENTS		7.1 Library form	23
Nescription of the language	Introduction	3	7.2 Import and export levels	25
1 Overview of Scheme			7.3 Examples	26
1.1 Basic types 5 8.2 Top-level program semantics 28 1.2 Expressions 6 9 Primitive syntax 28 1.3 Variables and binding 6 9.1 Primitive expression types 28 1.4 Definitions 6 9.2 Macros 29 1.5 Forms 7 10 Expansion process 29 1.6 Procedures 7 11.1 Base library 31 1.7 Procedure calls and syntactic keywords 7 11.1 Base types 31 1.8 Assignment 7 11.2 Definitions 31 1.9 Derived forms and macros 8 11.3 Booties 32 1.10 Syntactic data and datum values 8 11.4 Expressions 32 1.11 Continuations 8 11.5 Equivalence predicates 37 1.12 Libraries 9 11.6 Procedure predicate 39 1.13 Top-level programs 9 11.6 Procedure predicate 39 1.2 Libraries 9 11.7 Arithmetic 39 2 Requirement levels 9 11.8 Booleans 47 3.1 Numerical tower 10 11.10Symbols 49 3.2 Exactness		_	8 Top-level programs	27
1.2 Expressions 6 9 Primitive syntax 28 1.3 Variables and binding 6 9.1 Primitive expression types 28 1.4 Definitions 6 9.2 Macros 29 1.5 Forms 7 10 Expansion process 29 1.6 Procedures 7 11 Base library 31 1.7 Procedure calls and syntactic keywords 7 11.1 Base types 31 1.8 Assignment 7 11.2 Definitions 31 1.9 Derived forms and macros 8 11.3 Bodies 32 1.10 Syntactic data and datum values 8 11.4 Expressions 32 1.11 Continuations 8 11.5 Equivalence predicates 37 1.12 Libraries 9 11.6 Procedure predicate 39 1.13 Top-level programs 9 11.6 Procedure predicate 39 1.13 Numbers 10 11.9 Pairs and lists 47 3 Numbers 10 11.10 Symbols 49 3.2 Exactness 10 11.11 Characters 50 3.3 Fixnums and flonums 10 11.12 Strings 50 3.4 Implementation requirements		_	8.1 Top-level program syntax	27
1.3 Variables and binding 6 9.1 Primitive expression types 28 1.4 Definitions 6 9.2 Macros 29 1.5 Forms 7 10 Expansion process 29 1.6 Procedures 7 11. Base library 31 1.7 Procedure calls and syntactic keywords 7 11.1 Base types 31 1.8 Assignment 7 11.2 Definitions 31 1.9 Derived forms and macros 8 11.3 Bodies 32 1.10 Syntactic data and datum values 8 11.3 Bodies 32 1.10 Continuations 8 11.5 Equivalence predicates 32 1.11 Continuations 8 11.5 Equivalence predicates 37 1.12 Libraries 9 11.6 Procedure predicate 39 1.13 Top-level programs 9 11.6 Procedure predicate 39 1.14 Kinnes 9 11.7 Arithmetic 39 2 Requirement levels 9 11.8 Booleans 47 3 Numbers<		_		28
1.4 Definitions 6 9.2 Macros 29 1.5 Forms 7 10 Expansion process 29 1.6 Procedures 7 11 Base library 31 1.7 Procedure calls and syntactic keywords 7 11.1 Base types 31 1.8 Assignment 7 11.2 Definitions 31 1.9 Derived forms and macros 8 11.3 Bodies 32 1.10 Syntactic data and datum values 8 11.4 Expressions 32 1.11 Continuations 8 11.5 Equivalence predicates 37 1.12 Libraries 9 11.6 Procedure predicate 39 1.13 Top-level programs 9 11.7 Arithmetic 39 1.13 Top-level programs 9 11.8 Booleans 47 3 Numbers 10 11.9 Booleans 47 3 Numbers 10 11.10 Symbols 49 3.2 Exactness 10 11.10 Symbols 49 3.2 Exactness 10 11.11 Symbols 50 </td <td></td> <td></td> <td>9 Primitive syntax</td> <td>28</td>			9 Primitive syntax	28
1.5 Forms 7 10 Expansion process 29 1.6 Procedures 7 11 Base library 31 1.7 Procedure calls and syntactic keywords 7 11.1 Base types 31 1.8 Assignment 7 11.2 Definitions 31 1.9 Derived forms and macros 8 11.3 Bodies 32 1.10 Syntactic data and datum values 8 11.4 Expressions 32 1.11 Continuations 8 11.5 Equivalence predicates 37 1.12 Libraries 9 11.6 Procedure predicate 39 1.13 Top-level programs 9 11.7 Arithmetic 39 2 Requirement levels 9 11.8 Booleans 47 3 Numbers 10 11.9 Pairs and lists 47 3.1 Numerical tower 10 11.10Symbols 49 3.2 Exactness 10 11.11Characters 50 3.3 Fixnums and flonums 10 11.12Strings 50 3.4 Implementation requirements 10 11.1		6	9.1 Primitive expression types	28
1.6 Procedures 7 11 Base library 31 1.7 Procedure calls and syntactic keywords 7 11.1 Base types 31 1.8 Assignment 7 11.2 Definitions 31 1.9 Derived forms and macros 8 11.3 Bodies 32 1.10 Syntactic data and datum values 8 11.4 Expressions 32 1.11 Continuations 8 11.5 Equivalence predicates 37 1.12 Libraries 9 11.6 Procedure predicate 39 1.13 Top-level programs 9 11.7 Arithmetic 39 2 Requirement levels 9 11.8 Booleans 47 3 Numbers 10 11.9 Pairs and lists 47 3.1 Numerical tower 10 11.10Symbols 49 3.2 Exactness 10 11.11Characters 50 3.3 Fixums and flonums 10 11.12Strings 50 3.4 Implementation requirements 10 11.13Vectors 51 3.5 Infinities and NaNs 11 <t< td=""><td>1.4 Definitions</td><td>6</td><td>9.2 Macros</td><td>29</td></t<>	1.4 Definitions	6	9.2 Macros	29
1.7 Procedure calls and syntactic keywords 7 11.1 Base types 31 1.8 Assignment 7 11.2 Definitions 31 1.9 Derived forms and macros 8 11.3 Bodies 32 1.10 Syntactic data and datum values 8 11.4 Expressions 32 1.11 Continuations 8 11.5 Equivalence predicates 37 1.12 Libraries 9 11.6 Procedure predicate 39 1.13 Top-level programs 9 11.7 Arithmetic 39 2 Requirement levels 9 11.8 Booleans 47 3 Numbers 10 11.9 Pairs and lists 47 3.1 Numerical tower 10 11.10Symbols 49 3.2 Exactness 10 11.11Characters 50 3.3 Fixnums and flonums 10 11.12Strings 50 3.4 Implementation requirements 10 11.13Vectors 51 3.5 Infinities and NaNs 11 11.14Errors and violations 52 3.6 Distinguished -0.0	1.5 Forms	7	10 Expansion process	29
1.8 Assignment 7 11.2 Definitions 31 1.9 Derived forms and macros 8 11.3 Bodies 32 1.10 Syntactic data and datum values 8 11.4 Expressions 32 1.11 Continuations 8 11.5 Equivalence predicates 37 1.12 Libraries 9 11.6 Procedure predicate 39 1.13 Top-level programs 9 11.7 Arithmetic 39 2 Requirement levels 9 11.8 Booleans 47 3 Numbers 10 11.9 Pairs and lists 47 3.1 Numerical tower 10 11.10Symbols 49 3.2 Exactness 10 11.11Characters 50 3.4 Implementation requirements 10 11.12Strings 50 3.4 Implementation requirements 10 11.13Vectors 51 3.5 Infinities and NaNs 11 11.14Errors and violations 52 3.6 Distinguished -0.0 11 11.15Control features 53 4 Lexical syntax and datum syntax 11 11.16Iteration 55 4.1 Notation 12 11.17Quasiquotation 55 4.2 Lexical	1.6 Procedures	7	11 Base library	31
1.9 Derived forms and macros 8 11.3 Bodies 32 1.10 Syntactic data and datum values 8 11.4 Expressions 32 1.11 Continuations 8 11.5 Equivalence predicates 37 1.12 Libraries 9 11.6 Procedure predicate 39 1.13 Top-level programs 9 11.7 Arithmetic 39 2 Requirement levels 9 11.8 Booleans 47 3 Numbers 10 11.9 Pairs and lists 47 3.1 Numerical tower 10 11.10Symbols 49 3.2 Exactness 10 11.11Characters 50 3.3 Fixnums and flonums 10 11.12Strings 50 3.4 Implementation requirements 10 11.13Vectors 51 3.5 Infinities and NaNs 11 11.14Errors and violations 52 3.6 Distinguished -0.0 11 11.15Control features 53 4 Lexical syntax and datum syntax 11 11.16Iteration 55 4.1 Notation 12 11.17Quasiquotation 55 4.2 Lexical syntax 12 11.18Binding constructs for syntactic keywords 56	1.7 Procedure calls and syntactic keywords	7	11.1 Base types	31
1.10 Syntactic data and datum values 8 11.4 Expressions 32 1.11 Continuations 8 11.5 Equivalence predicates 37 1.12 Libraries 9 11.6 Procedure predicate 39 1.13 Top-level programs 9 11.7 Arithmetic 39 2 Requirement levels 9 11.8 Booleans 47 3 Numbers 10 11.9 Pairs and lists 47 3.1 Numerical tower 10 11.10Symbols 49 3.2 Exactness 10 11.11Characters 50 3.3 Fixnums and flonums 10 11.12Strings 50 3.4 Implementation requirements 10 11.13Vectors 51 3.5 Infinities and NaNs 11 11.4Errors and violations 52 3.6 Distinguished -0.0 11 11.15Control features 53 4 Lexical syntax and datum syntax 11 11.16Iteration 55 4.1 Notation 12 11.17Quasiquotation 55 4.2 Lexical syntax 12 11.18Binding constructs for syntactic keywords 56 4.3 Datum syntax 16 11.19Macro transformers 57	1.8 Assignment	7	11.2 Definitions	31
1.11 Continuations 8 11.5 Equivalence predicates 37 1.12 Libraries 9 11.6 Procedure predicate 39 1.13 Top-level programs 9 11.7 Arithmetic 39 2 Requirement levels 9 11.8 Booleans 47 3 Numbers 10 11.9 Pairs and lists 47 3.1 Numerical tower 10 11.10Symbols 49 3.2 Exactness 10 11.11Characters 50 3.3 Fixnums and flonums 10 11.12Strings 50 3.4 Implementation requirements 10 11.13Vectors 51 3.5 Infinities and NaNs 11 11.4Errors and violations 52 3.6 Distinguished -0.0 11 11.5Control features 53 4 Lexical syntax and datum syntax 11 11.16Iteration 55 4.1 Notation 12 11.17Quasiquotation 55 4.2 Lexical syntax 12 11.18Binding constructs for syntactic keywords 56 4.3 Datum syntax 16 11.19Macro transformers 57 5 Semantic concepts 17 11.20Tail calls and tail contexts 59 <	1.9 Derived forms and macros	8	11.3 Bodies	32
1.12 Libraries 9 11.6 Procedure predicate 39 1.13 Top-level programs 9 11.7 Arithmetic 39 2 Requirement levels 9 11.8 Booleans 47 3 Numbers 10 11.9 Pairs and lists 47 3.1 Numerical tower 10 11.10Symbols 49 3.2 Exactness 10 11.11Characters 50 3.3 Fixnums and flonums 10 11.12Strings 50 3.4 Implementation requirements 10 11.13Vectors 51 3.5 Infinities and NaNs 11 11.14Errors and violations 52 3.6 Distinguished -0.0 11 11.15Control features 53 4 Lexical syntax and datum syntax 11 11.16Iteration 55 4.1 Notation 12 11.17Quasiquotation 55 4.2 Lexical syntax 12 11.18Binding constructs for syntactic keywords 56 4.3 Datum syntax 16 11.19Macro transformers 57 5 Semantic concepts 17 11.20Tail calls and tail contexts 59	1.10 Syntactic data and datum values	8	11.4 Expressions	32
1.13 Top-level programs 9 11.7 Arithmetic 39 2 Requirement levels 9 11.8 Booleans 47 3 Numbers 10 11.9 Pairs and lists 47 3.1 Numerical tower 10 11.10Symbols 49 3.2 Exactness 10 11.11Characters 50 3.3 Fixnums and flonums 10 11.12Strings 50 3.4 Implementation requirements 10 11.13Vectors 51 3.5 Infinities and NaNs 11 11.14Errors and violations 52 3.6 Distinguished -0.0 11 11.15Control features 53 4 Lexical syntax and datum syntax 11 11.16Iteration 55 4.1 Notation 12 11.17Quasiquotation 55 4.2 Lexical syntax 12 11.18Binding constructs for syntactic keywords 56 4.3 Datum syntax 16 11.19Macro transformers 57 5 Semantic concepts 17 11.20Tail calls and tail contexts 59	1.11 Continuations	8	11.5 Equivalence predicates	37
1.13 Top-level programs 9 11.7 Arithmetic 39 2 Requirement levels 9 11.8 Booleans 47 3 Numbers 10 11.9 Pairs and lists 47 3.1 Numerical tower 10 11.10Symbols 49 3.2 Exactness 10 11.11Characters 50 3.3 Fixnums and flonums 10 11.12Strings 50 3.4 Implementation requirements 10 11.13Vectors 51 3.5 Infinities and NaNs 11 11.14Errors and violations 52 3.6 Distinguished -0.0 11 11.15Control features 53 4 Lexical syntax and datum syntax 11 11.16Iteration 55 4.1 Notation 12 11.17Quasiquotation 55 4.2 Lexical syntax 12 11.18Binding constructs for syntactic keywords 56 4.3 Datum syntax 16 11.19Macro transformers 57 5 Semantic concepts 17 11.20Tail calls and tail contexts 59	1.12 Libraries	9	11.6 Procedure predicate	39
3 Numbers 10 11.9 Pairs and lists 47 3.1 Numerical tower 10 11.10Symbols 49 3.2 Exactness 10 11.11Characters 50 3.3 Fixnums and flonums 10 11.12Strings 50 3.4 Implementation requirements 10 11.13Vectors 51 3.5 Infinities and NaNs 11 11.14Errors and violations 52 3.6 Distinguished -0.0 11 11.15Control features 53 4 Lexical syntax and datum syntax 11 11.16Iteration 55 4.1 Notation 12 11.17Quasiquotation 55 4.2 Lexical syntax 12 11.18Binding constructs for syntactic keywords 56 4.3 Datum syntax 16 11.19Macro transformers 57 5 Semantic concepts 17 11.20Tail calls and tail contexts 59	1.13 Top-level programs	9		39
3 Numbers 10 11.9 Pairs and lists 47 3.1 Numerical tower 10 11.10Symbols 49 3.2 Exactness 10 11.11Characters 50 3.3 Fixnums and flonums 10 11.12Strings 50 3.4 Implementation requirements 10 11.13Vectors 51 3.5 Infinities and NaNs 11 11.14Errors and violations 52 3.6 Distinguished -0.0 11 11.15Control features 53 4 Lexical syntax and datum syntax 11 11.16Iteration 55 4.1 Notation 12 11.17Quasiquotation 55 4.2 Lexical syntax 12 11.18Binding constructs for syntactic keywords 56 4.3 Datum syntax 16 11.19Macro transformers 57 5 Semantic concepts 17 11.20Tail calls and tail contexts 59		9	11.8 Booleans	47
3.2 Exactness 10 11.11Characters 50 3.3 Fixnums and flonums 10 11.12Strings 50 3.4 Implementation requirements 10 11.13Vectors 51 3.5 Infinities and NaNs 11 11.14Errors and violations 52 3.6 Distinguished -0.0 11 11.15Control features 53 4 Lexical syntax and datum syntax 11 11.16Iteration 55 4.1 Notation 12 11.17Quasiquotation 55 4.2 Lexical syntax 12 11.18Binding constructs for syntactic keywords 56 4.3 Datum syntax 16 11.19Macro transformers 57 5 Semantic concepts 17 11.20Tail calls and tail contexts 59	9 Numbers	10	11.9 Pairs and lists	47
3.3 Fixnums and flonums 10 11.12Strings 50 3.4 Implementation requirements 10 11.13Vectors 51 3.5 Infinities and NaNs 11 11.14Errors and violations 52 3.6 Distinguished -0.0 11 11.15Control features 53 4 Lexical syntax and datum syntax 11 11.16Iteration 55 4.1 Notation 12 11.17Quasiquotation 55 4.2 Lexical syntax 12 11.18Binding constructs for syntactic keywords 56 4.3 Datum syntax 16 11.19Macro transformers 57 5 Semantic concepts 17 11.20Tail calls and tail contexts 59	3.1 Numerical tower	10	11.10Symbols	49
3.4 Implementation requirements 10 11.13 Vectors 51 3.5 Infinities and NaNs 11 11.14 Errors and violations 52 3.6 Distinguished -0.0 11 11.15 Control features 53 4 Lexical syntax and datum syntax 11 11.16 Iteration 55 4.1 Notation 12 11.17 Quasiquotation 55 4.2 Lexical syntax 12 11.18 Binding constructs for syntactic keywords 56 4.3 Datum syntax 16 11.19 Macro transformers 57 5 Semantic concepts 17 11.20 Tail calls and tail contexts 59	3.2 Exactness	10	11.11Characters	50
3.5 Infinities and NaNs 11 11.14Errors and violations 52 3.6 Distinguished -0.0 11 11.15Control features 53 4 Lexical syntax and datum syntax 11 11.16Iteration 55 4.1 Notation 12 11.17Quasiquotation 55 4.2 Lexical syntax 12 11.18Binding constructs for syntactic keywords 56 4.3 Datum syntax 16 11.19Macro transformers 57 5 Semantic concepts 17 11.20Tail calls and tail contexts 59	3.3 Fixnums and flonums	10	11.12Strings	50
3.5 Infinities and NaNs 11 11.14Errors and violations 52 3.6 Distinguished -0.0 11 11.15Control features 53 4 Lexical syntax and datum syntax 11 11.16Iteration 55 4.1 Notation 12 11.17Quasiquotation 55 4.2 Lexical syntax 12 11.18Binding constructs for syntactic keywords 56 4.3 Datum syntax 16 11.19Macro transformers 57 5 Semantic concepts 17 11.20Tail calls and tail contexts 59	3.4 Implementation requirements	10	11.13Vectors	51
3.6 Distinguished -0.0 11 11.15Control features 53 4 Lexical syntax and datum syntax 11 11.16Iteration 55 4.1 Notation 12 11.17Quasiquotation 55 4.2 Lexical syntax 12 11.18Binding constructs for syntactic keywords 56 4.3 Datum syntax 16 11.19Macro transformers 57 5 Semantic concepts 17 11.20Tail calls and tail contexts 59		11	11.14Errors and violations	52
4 Lexical syntax and datum syntax 11 11.16Iteration 55 4.1 Notation 12 11.17Quasiquotation 55 4.2 Lexical syntax 12 11.18Binding constructs for syntactic keywords 56 4.3 Datum syntax 16 11.19Macro transformers 57 5 Semantic concepts 17 11.20Tail calls and tail contexts 59	3.6 Distinguished -0.0	11	11.15Control features	53
4.1 Notation 12 11.17Quasiquotation 55 4.2 Lexical syntax 12 11.18Binding constructs for syntactic keywords 56 4.3 Datum syntax 16 11.19Macro transformers 57 5 Semantic concepts 17 11.20Tail calls and tail contexts 59		11	11.16Iteration	55
4.2 Lexical syntax		12	11.17Quasiquotation	55
4.3 Datum syntax		12	11.18Binding constructs for syntactic keywords .	56
5 Semantic concepts		16	11.19Macro transformers	57
		17	11.20Tail calls and tail contexts	59
5.1 Programs and libraries	5.1 Programs and libraries	17	Appendices	
5.2 Variables, keywords, and regions				61
5.3 Exceptional situations				

That's odd...



Numerical languages are strangely diverse.

General languages are strangely numerical.

What's going on?

The "niche hypothesis"



Numerical computing is still an under-generalized niche

- each language picks a different view of how to specialize
- also the oldest programming language niche Fortran

Hypothesis:

- many diverse languages in this niche can be replaced
- by a single sufficiently powerful general-purpose language

History



Text processing was a niche with a similar variety of languages

SNOBOL, SPITBOL, COMIT, TRAC, TTM, Icon, Unicon, sed, awk, Perl4 lot's of different views of text processing and how to specialize for it

You don't see much of these anymore

people use one of Python, Ruby, or Perl5 instead
 (we still use sed and awk sometimes, but we could use Perl/Python/Ruby)

Diverse languages specialized in aspects of text processing

replaced by general languages that support it instead

History



Text processing was diverse & hard.

Now it's unified & easy.

Can we do this for numerical computing?

History



We believe the answer is "yes"

Julia is our attempt to do this.

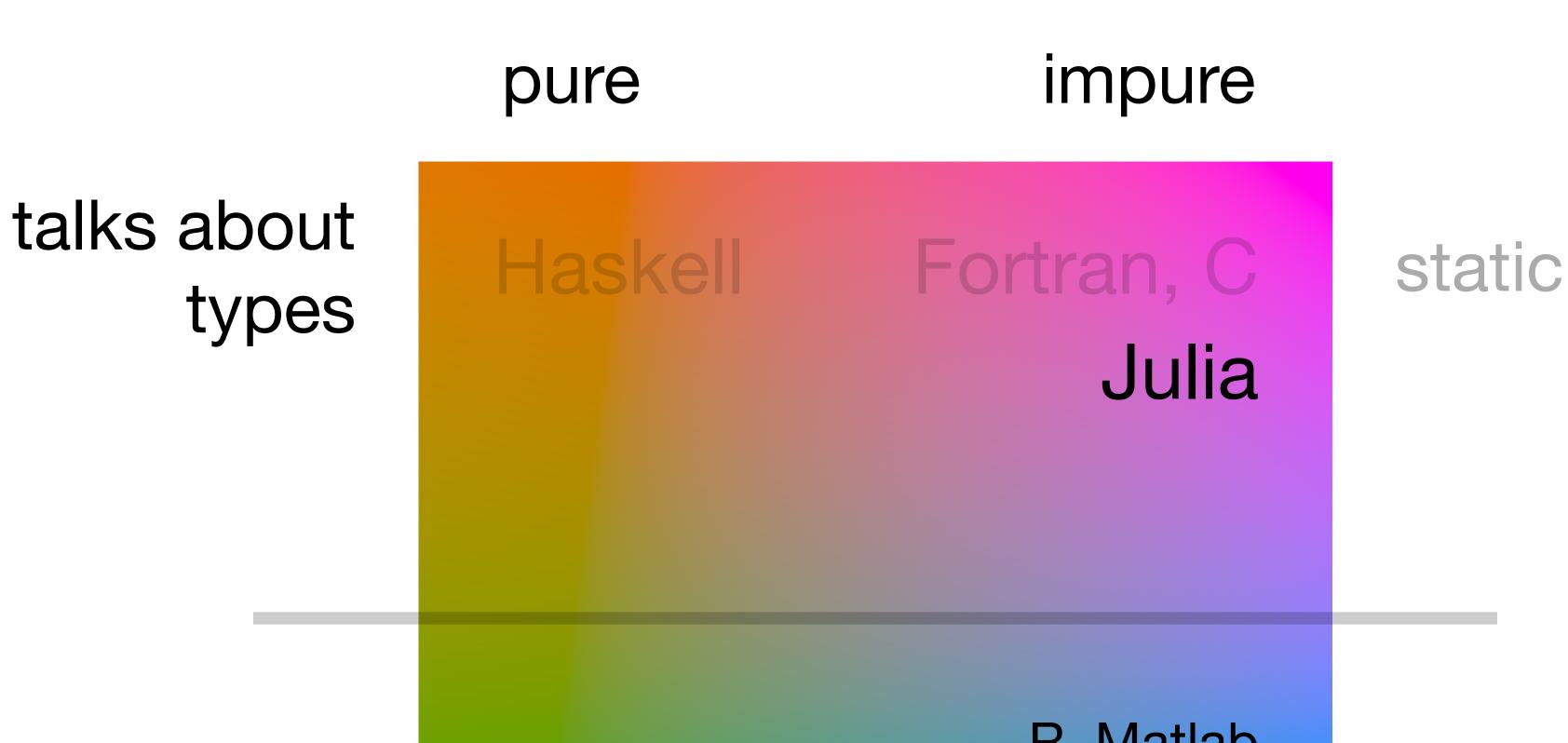
One principle



Stop making numbers special.

In Julia, all numeric types are user-defined.

Some of them are just defined for you.



doesn't talk about types

λ-calculus

R, Matlab Mathematica Python ...

Lisp

dynamic