Physical Units Expression Language

PUEL Specification for SciNumTools v3

Ondrej Pego Jaura

September 2026

Abstract

This specification defines the syntax and semantics of the Physical Units Expression Language (PUEL), a compact notation for physical quantities, unit expressions, uncertainties, and system-specific entities in SciNumTools3. It provides the standalone expression grammar, validity and conformance requirements, uncertainty rules, special-unit behavior, and reference tables for supported unit systems. PUEL is designed for use in scientific software and machine-readable exchange; the specification also identifies a portable profile for interoperable expressions.

Contents

1 PUEL Specification **3**
- 1.1 Overview . . . 3
  - 1.1.1 Terminology . . . 3
- 1.2 Design Principles . . . 3
  - 1.2.1 Relationship to Existing Unit Standards . . . 3
  - 1.2.2 Portable PUEL profile . . . 3
- 1.3 Unit Model . . . 4
  - 1.3.1 Base units . . . 4
  - 1.3.2 Scaling factors . . . 4
  - 1.3.3 Named Entities . . . 4
  - 1.3.4 Custom units . . . 5
- 1.4 Unit expressions . . . 5
  - 1.4.1 General Form . . . 5
  - 1.4.2 Expression Parsing . . . 6
  - 1.4.3 Numerical arrays . . . 6
  - 1.4.4 Unit-system selection . . . 6
  - 1.4.5 Display notation . . . 6
  - 1.4.6 Normalization and Serialization . . . 6
- 1.5 Calculator expressions . . . 7
- 1.6 Conversions . . . 7
  - 1.6.1 Logarithmic and temperature conversions . . . 7
  - 1.6.2 Conversion between unit systems . . . 7

2 Validity and Errors **7**
- 2.1 Lexical and structural validity . . . 8
- 2.2 Vocabulary and system validity . . . 8
- 2.3 Dimensional and shape validity . . . 8
- 2.4 Diagnostics . . . 8

3 Uncertainty Propagation **8**
- 3.1 Reference PUQ propagation model . . . 8

4 Special Units and Conformance Options **9**
- 4.1 Temperature units . . . 9
- 4.2 Logarithmic units . . . 9
- 4.3 Conversion availability . . . 10
- 4.4 Reference implementation feature matrix . . . 10

5 Scaling prefixes **10**

6 Unit systems **11**
6.1 Reading the tables . . . 11

7 Shared unit definitions **11**
7.1 Base-unit symbols . . . 11
7.2 Fixed reference constants . . . 11

8 SI: International System of Units **12**
8.1 Derived units and aliases . . . 12
8.2 System-specific constants . . . 14
8.3 Quantity symbols and conversion factors . . . 15

9 ESU: Electrostatic CGS units **17**
9.1 Derived units and aliases . . . 17
9.2 System-specific constants . . . 17
9.3 Quantity symbols and conversion factors . . . 17

10 EMU: Electromagnetic CGS units **18**
10.1 Derived units and aliases . . . 18
10.2 System-specific constants . . . 19
10.3 Quantity symbols and conversion factors . . . 19

11 GU: Gaussian CGS units **20**
11.1 Derived units and aliases . . . 20
11.2 System-specific constants . . . 20
11.3 Quantity symbols and conversion factors . . . 21

12 IU: Imperial units **21**
12.1 Derived units and aliases . . . 21
12.2 Quantity symbols and conversion factors . . . 23

13 US: United States customary units **23**
13.1 Derived units and aliases . . . 23
13.2 Quantity symbols and conversion factors . . . 24

14 AU: Atomic units **25**
14.1 Derived units and aliases . . . 25
14.2 System-specific constants . . . 25
14.3 Quantity symbols and conversion factors . . . 25

15 PU: Planck units **26**
15.1 Derived units and aliases . . . 26
15.2 System-specific constants . . . 26
15.3 Quantity symbols and conversion factors . . . 26

16 SRU: Units for special relativity **27**
16.1 Derived units and aliases . . . 27
16.2 System-specific constants . . . 27
16.3 Quantity symbols and conversion factors . . . 27

17 GRU: Units for general relativity **27**
17.1 Derived units and aliases . . . 27
17.2 System-specific constants . . . 27
17.3 Quantity symbols and conversion factors . . . 28

18 GEO: Geometrized units **28**
18.1 Derived units and aliases . . . 28
18.2 System-specific constants . . . 28
18.3 Quantity symbols and conversion factors . . . 28

A Formal Grammar **29**

1 PUEL Specification

1.1 Overview

PUEL (Physical Units Expression Language) is a domain-specific language for representing and manipulating physical units and dimensional expressions.

It defines the syntax and structure of unit expressions used in scientific computations within the SciNumTools3 framework. Evaluation, normalization, dimensional analysis, and unit conversion are performed by the underlying EXS and PUQ modules.

The language is designed for the following use cases:

- As a component of DIPL for defining and validating unit-aware parameters.
- As a standalone language for the concise and machine-readable representation of physical unit expressions.
- As a backend for programmatic interfaces, such as Python bindings and other language integrations.

This chapter defines the language syntax and conversion rules. The following chapters provide reference tables for scaling prefixes, unit systems, and shared definitions, followed by the derived units, constants, and quantity scales of each supported system.

Focused normative sections define validity and errors, uncertainty propagation, and special-unit semantics and conformance options. The formal grammar is reproduced in the appendix from the maintained EBNF source.

1.1.1 Terminology

A **unit expression** describes units or dimensions, such as `kg*m*s-2`. A **quantity expression** additionally supplies a value, such as `12*kg*m*s-2`. Both are standalone, whitespace-free PUEL expressions. A **calculator expression**, such as `12*kg + 3*g`, is a PUQ operation over complete PUEL quantities and is not part of the standalone PUEL grammar. Host APIs may also construct quantities from numerical arrays or values without first parsing a PUEL literal.

1.2 Design Principles

PUEL is designed to be:

- **Concise** — compact textual representation of units
- **Composable** — units can be combined using algebraic operations
- **Deterministic** — expressions resolve to an unambiguous dimensional meaning
- **Extensible** — supports custom units and prefixes
- **Solver-backed** — evaluation delegated to EXS

1.2.1 Relationship to Existing Unit Standards

PUEL is a machine-readable language for representing physical quantities and unit expressions, rather than a replacement for established unit coding systems. Its vocabulary and dimensional model are designed to be understandable alongside SI, UCUM, UDUNITS, QUDT, and the OM ontology. PUEL additionally expresses numerical values, uncertainties, physical constants, symbolic quantities, and algebraic unit factors.

SciNumTools3 does not currently claim complete bidirectional conversion to those external serializations. A PUEL expression remains semantically independent of any particular external notation or ontology.

1.2.2 Portable PUEL profile

For exchange between implementations, producers SHOULD use the portable profile: ASCII syntax, SI linear units, no custom symbols, no system-specific constants or quantity scales, and no temperature or logarithmic units. Use compact array literals without spaces, such as `[2,3,4]*m`, when arrays are needed. Producers using an extended feature SHOULD record the required unit system and feature family with the exchanged data.

1.3 Unit Model

1.3.1 Base units

All unit expressions are ultimately represented in terms of a fixed set of base dimensions:

- Length in meters (`m`)
- Mass in grams (`g`)
- Time in seconds (`s`)
- Temperature in kelvins (`K`)
- Electric current in amperes (`A`)
- Luminous intensity in candelas (`cd`)
- Amount of substance in moles (`mol`)
- Angle in radians (`rad`)

Derived units are represented as combinations of these base dimensions together with an associated scaling factor.

The choice of base dimensions is driven by implementation considerations and does not exactly match the SI base-unit system.

- Mass is represented using grams (`g`) rather than kilograms (`kg`) to avoid embedding a prefix in a base unit.
- Electric current is represented using amperes (`A`); electric charge is a derived quantity (`C` = `A*s`).
- Angle is included as a base dimension to enable dimensional validation of mathematical functions that require angular arguments. However, angular units MUST remain convertible to dimensionless units, reflecting their dimensionless nature in the SI system.

1.3.2 Scaling factors

Scaling prefixes provide a compact way to represent decimal multiples and submultiples of units. A scaling prefix modifies a unit identifier by applying a fixed numerical factor while preserving the unit's dimensionality.

For example, the expressions `m`, `cm`, and `km` all represent units of length and differ only in their scaling factors relative to the base unit.

PUEL defines a standard set of scaling prefixes. Implementations MUST recognize and correctly apply these prefixes when parsing unit expressions. However, prefixes SHOULD only be permitted for units for which their use is conventional or widely accepted.

The complete list of supported scaling prefixes is provided in Scaling Prefixes.

1.3.3 Named Entities

Named entities in PUEL include all units (e.g., Joule `J`, degree Celsius `Cel`, decibel `dB`), physical constants (e.g., fixed proton mass `{#m_p}`, system-specific Avogadro constant `{N_A}`, solar luminosity `{L_sol}`), physical quantities (e.g., energy `<E>`, electric flux `<Phi_E>`, radiation dose `<D_r>`), and unit-system scaling factors (e.g., `|E|`, `|Phi_E|`, `|D_r|`). The dimensionality of these entities is defined in terms of the base dimensions described in the previous section.

PUEL distinguishes between these categories using the following notation:

- Units are represented by their symbols: `J`, `Cel`, `dB`
- Physical constants are enclosed in curly braces. A leading `#` denotes a fixed reference constant, such as `{#m_p}`; without it, the constant is resolved in the active unit system, such as `{N_A}` or `{L_sol}`.
- Physical quantities are enclosed in angle brackets: `<E>`, `<Phi_E>`, `<D_r>`
- Unit-system scaling factors are enclosed in pipe symbols: `|E|`, `|Phi_E|`, `|D_r|`

Unit-system scaling factors represent the numerical coefficients required to convert a physical quantity between the SI system and another system of units. They are primarily intended to support conversions between systems with different dimensional conventions, such as SI and CGS.

Each implementation MUST define the numerical values and dimensional representations of all supported derived entities. These definitions SHOULD be precomputed and stored to avoid repeated evaluation and dimensional derivation during the processing of unit expressions.

Within a given implementation and unit system, all supported identifiers and scaling-prefix combinations MUST be unique. Consequently, every valid token in a PUEL expression MUST have exactly one interpretation.

1.3.4 Custom units

Custom units MAY be defined in the surrounding environment using existing unit expressions. Once registered in a unit system, a custom symbol can be used in expressions just like a built-in unit, including in products, quotients, and powers. For example, an environment may define `step` as `0.75*m` and then interpret `4*step` as `3*m`.

Custom definitions are scoped to their unit system. PUQ stores their numerical scale, uncertainty when present, and dimensions. Registration is performed by the host application; PUEL does not define assignment syntax. PUQ custom units do not accept scaling prefixes.

1.4 Unit expressions

1.4.1 General Form

A unit expression consists of an optional unit system identifier followed by a numerical value, unit factors, or both.

General form:

`[<system>_][<value>*][<unit-expression>]`

where:

- `<system>` is a unit system identifier.
- `<value>` is a numerical value, optionally including an uncertainty.
- `<unit-expression>` is a combination of unit identifiers and algebraic operators.

At least one of `<value>` or `<unit-expression>` MUST be present.

The `*` separator between the value and unit factors is present only when both are supplied. A value alone, such as `2.5`, represents a dimensionless quantity. The value MAY also be a numerical array, as described below.

Examples:

```
m
2.5*m
3.452(3)*kg*m/s2
US_lb*ft
SI_9.81*m/s2
```

The following rules apply in unit expressions:

Rule	Examples
Expressions MUST NOT contain whitespace. PUQ MAY accept one space after an array comma as an input convenience, but portable and canonical array literals omit it.	`kg*m/s2`; `[2,3,4]*m`
Unit identifiers MAY be combined using multiplication (`*`), division (`/`), and grouping parentheses (`(`, `)`).	`kg*m`; `m/s`; `kg*m2/(sr*s2)`
Exponents are written directly after the corresponding unit identifier and do not use exponentiation operators such as `^` or `**`.	`m2`
Fractional exponents are written using the `numerator:denominator` notation.	`kg3:2`
Negative exponents are written using a leading minus sign.	`s-1`
A unit system MAY be specified. When present, it MUST appear at the beginning of the expression and MUST be separated from the remainder by an underscore (`_`).	`US_lb*ft`
Numerical values MAY be included in expressions.	`2.34e3*mol`; `-9.81*m/s2`
The symbols `+` and `-` denote the sign of a numerical value or exponent; they do not represent addition or subtraction operators.	`+5*m`; `-5*m`; `s-2`
Uncertainties are written in parentheses immediately following the last decimal digit of a numerical value. The number of digits in the uncertainty corresponds to the same number of least significant digits in the value.	`3.45234(2)e3` → `3452.34 ± 0.02`; `3.45234(12)` → `3.45234 ± 0.00012`; `12.3(4)e-2` → `0.123 ± 0.004`; `120(5)` → `120 ± 5`

1.4.2 Expression Parsing

Although numerical values are typically written at the beginning of an expression, they are not restricted to that position. Numerical factors MAY appear anywhere within a unit expression as multiplicative constants. For example, `2*kg*m/s2` and `kg*(2*m)/s2` are equivalent.

Implementations SHOULD accept numerical factors in any valid position within a unit expression. However, producers of unit expressions SHOULD place any numerical factor at the beginning of the expression, as this improves readability and simplifies parsing.

Unit expressions MAY use division operators and grouping parentheses for readability. For example, `kg*m2/(sr*s2)` and `kg*m2*sr-1*s-2` represent the same unit expression.

An exponent MAY follow a closing parenthesis and applies to the entire grouped expression, including numerical factors. Integer, negative, and fractional exponents use the same notation as unit exponents. For example, `(m/s)2` is equivalent to `m2*s-2`, and `kg/(m*s)-1:2` is equivalent to `kg*m1:2*s1:2`. Exponents also apply to named entities, as in `kg*<v>2`.

Expressions containing only numerical factors are supported. For example, `3*(2.0e1/5.0)` evaluates to the dimensionless value `12`.

1.4.3 Numerical arrays

A numerical array is written as a comma-separated list of values in square brackets. All elements share the units specified outside the brackets. Elements MAY include scientific notation or parenthesized uncertainties. The portable form contains no spaces; PUQ also accepts one optional space after a comma for input compatibility.

```
[2,3.4,5e6]*km/s
[2.00(20),3.00(30)]*m
```

Multiplication and division by a scalar apply to each element. Multiplication and division between arrays of the same shape are elementwise operations. For example, `[20,40.5]*2` evaluates to `[40,81]`, and `[20,40.5]*[2,3]` evaluates to `[40,121.5]`. Unit factors are combined using the ordinary expression rules.

1.4.4 Unit-system selection

The system identifier, when present, MUST appear only at the beginning of the expression, immediately followed by an underscore (`_`). It selects the interpretation of all unit factors in the expression. For example, `US_23*ft2` is valid; `US 23*ft2` and `23*US_ft2` are not valid PUEL expressions. PUQ supports SI and, when enabled, CGS variants (`ESU`, `EMU`, `GU`), English systems (`IU`, `US`), and natural systems (`AU`, `PU`, `SRU`, `GRU`, `GEO`).

When no identifier is supplied, PUQ uses the active unit system of the surrounding environment, initially SI. If both the expression and its environment explicitly select a system for that expression, the selections MUST agree. Producers SHOULD include the system identifier when an expression must retain its interpretation outside its original environment.

1.4.5 Display notation

In addition to ASCII notation, PUQ can produce Unicode and HTML representations for display. When Unicode input preprocessing is enabled, it accepts superscript exponents, the Unicode middle dot (U+22C5) for multiplication, the Unicode minus sign (U+2212), and the multiplication sign (U+00D7) for scientific notation. For example, a displayed form using U+00D7 before `10` and superscript U+2212/U+22C5 symbols corresponds to the portable expression `6.23537e-2*C3`.

Unicode input conveniences are optional; ASCII notation remains the portable form. A system identifier MUST retain its leading position and underscore separator even when Unicode notation is used, as in the displayed form of `US_23*ft2`. HTML output is a display format, not an expression input format.

1.4.6 Normalization and Serialization

Implementations MUST resolve a valid expression to its numerical scale, dimensions, and applicable special-unit semantics. Division may be normalized to negative exponents; for example, `kg*m2/(sr*s2)` and `kg*m2*sr-1*s-2` have the same dimensional meaning.

PUEL does not yet prescribe one unique textual serialization for every semantically equivalent expression. Implementations may retain factor order or choose their own normalized display form. They MUST NOT imply that formatting alone preserves the original placement of division operators or grouping parentheses.

For portable serialization, producers SHOULD use ASCII notation, place a numerical factor first, join factors with `*`, use negative exponents rather than `/`, and omit whitespace from array literals. Implementations that offer a stronger canonical serialization MUST document their factor ordering, numeric precision, uncertainty formatting, prefix rebasing, and treatment of temperature and logarithmic units.

When representing base dimensions, only dimensions with non-zero exponents SHOULD be included. Their order MUST follow the base-unit list in Base units: `m`, `g`, `s`, `K`, `A`, `cd`, `mol`, and `rad`.

1.5 Calculator expressions

The grammar in `grammar.ebnf` defines a single, whitespace-free PUEL unit expression. `Quantity` and `UnitSolver` parse that form directly; for example, write `12*km`, not `12 km`.

PUQ also provides a calculator layer for arithmetic between complete PUEL quantities. Its binary operators are written with spaces around them, so each operand remains a valid PUEL expression:

```
23*cm + 3*m
2*g * ( 6*m2/s2 - 3*cm2 / 1.5*s2 )
```

The calculator supports grouping, unary `+` and `-`, and binary `+`, `-`, `*`, and `/`, in the usual precedence order. Calculator syntax is a PUQ host operation rather than an additional form accepted by the standalone PUEL unit solver. Likewise, the C++ `puq::math` functions such as `sqrt`, `pow`, `log`, and trigonometric functions operate on parsed quantities but do not add function-call syntax to PUEL.

1.6 Conversions

PUQ conversion APIs operate on quantities parsed from PUEL and can convert between units of identical dimensions:

`km` → `m`
`eV` → `J`
`dBW` → `W`

Special handling exists for:

- logarithmic units
- temperature units

These require custom conversion rules.

1.6.1 Logarithmic and temperature conversions

Logarithmic conversions distinguish ratios from levels with a defined reference. PUQ supports power ratios (`PR`), amplitude ratios (`AR`), and logarithmic units such as decibels (`dB`) and nepers (`Np`). Power levels such as `dBm` and `dBW` include their reference power: `0*dBm` corresponds to `1*mW`, while `0*dBW` corresponds to `1*W`. A bare `dB` value does not specify an absolute power.

Absolute temperature conversions can include an offset as well as a scale. For example, `23*Cel` converts to `296.15*K`. PUQ supports Celsius (`Cel`), Fahrenheit (`degF`), kelvin (`K`), and Rankine (`degR`). Such conversions cannot be described by a multiplicative scaling factor alone.

1.6.2 Conversion between unit systems

Conversions between systems with different dimensional conventions may require a physical-quantity context in addition to source and target units. For example, converting SI current in `A` to electrostatic CGS current in `statA` requires the current context `I`. Energy conversion between `J` and `erg` does not require an explicit context.

The context identifies the corresponding physical quantity in each system and enables the use of its unit-system scaling factors. It is supplied by the surrounding application, not appended as an additional operator in the expression. A system prefix selects a system; it does not by itself specify a conversion context.

2 Validity and Errors

This section defines when a PUEL expression is valid independently of the particular API used to parse it.

2.1 Lexical and structural validity

A standalone PUEL expression MUST conform to the grammar in `grammar.ebnf`. In particular:

- it MUST contain at least one value, unit factor, named entity, or grouped expression;
- whitespace is forbidden, including after an array comma;
- an array literal is one-dimensional and MUST contain at least one scalar value;
- a fractional exponent denominator MUST be non-zero;
- a system identifier, when present, MUST be the first token and MUST be followed by `_`; and
- `+` and `-` are signs only in standalone PUEL. Addition and subtraction belong to the spaced PUQ calculator syntax.

An implementation MUST reject malformed delimiters, unknown identifiers, invalid numerical literals, malformed uncertainties, and unbalanced grouping parentheses. It MUST NOT silently reinterpret an unknown identifier as a prefix-plus-unit combination.

2.2 Vocabulary and system validity

Each unit, prefix combination, constant, quantity scale, and system factor MUST resolve unambiguously in the selected unit system. A prefix is valid only when the selected unit explicitly permits that prefix. Custom symbols follow the same uniqueness rule and do not accept prefixes in the PUQ reference implementation. PUQ may accept an optional space after a literal array comma as a non-portable input convenience.

Without an explicit system identifier, resolution uses the active unit system. If an API separately supplies a system and the expression contains a different system identifier, the expression is invalid because its interpretation would be ambiguous.

2.3 Dimensional and shape validity

Parsing establishes dimensions but does not by itself request a conversion. Operations and conversions impose the following additional requirements:

- addition and subtraction require compatible dimensions;
- conversion requires compatible dimensions, unless a documented cross-system quantity context is supplied;
- a quantity used as an exponent in a PUQ mathematical operation MUST be dimensionless; and
- elementwise array operations require compatible array shapes.

The literal PUEL grammar deliberately defines only one-dimensional arrays. PUQ can also process multidimensional values supplied through the host VAL/C++ API; that capability is not a multidimensional PUEL literal syntax.

2.4 Diagnostics

Implementations SHOULD report the failing expression or unit, the reason it is invalid, and a corrective action. The PUQ reference implementation distinguishes parse failures, unknown or disallowed prefixes, unit-system conflicts, dimension mismatch, and unavailable special-unit conversions.

3 Uncertainty Propagation

A scalar PUEL uncertainty is written immediately after its estimate, for example `3.452(3)*kg`. The parenthesized digits represent an **absolute** uncertainty in the last significant digits of the estimate. Thus `3.452(3)` represents `3.452 ± 0.003`, while `3.452(3)e3` represents `3452 ± 3`.

Array literals may attach an uncertainty to each element:

```
[2.00(20),3.00(30)]*m
```

The estimate and uncertainty arrays MUST have the same shape. A quantity without parenthesized uncertainty has no stored uncertainty.

3.1 Reference PUQ propagation model

PUQ stores absolute uncertainty in the same units as the estimate. Linear unit conversion scales both estimate and uncertainty by the conversion factor. Offset temperature conversion changes the estimate by its offset while scaling the uncertainty only by the scale.

PUQ propagates standard uncertainties to first order. For addition and subtraction, it assumes the input uncertainties are independent and combines them by root-sum-square:

```
(x ± dx) ± (y ± dy) = (x ± y) ± sqrt(dx² + dy²)
```

For multiplication and division, PUQ uses first-order derivatives and combines the independent contributions by root-sum-square. For example:

```
u(xy) = sqrt((y·u(x))² + (x·u(y))²)
u(x/y) = sqrt((u(x)/y)² + (x·u(y)/y²)²)
```

The public PUQ mathematical functions use first-order derivative propagation where applicable. For example, for a constant exponent `p`:

```
y = x^p
dy = |p * x^(p - 1)| * dx
```

For a variable exponent, the independent contributions are combined by root-sum-square. Logarithmic conversion and mathematical functions propagate uncertainty through their corresponding transformation.

By default, inputs are independent. Copies and differentiable operations retain source identity, so reusing one uncertain quantity preserves its correlation (for example, `x - x` has zero uncertainty). Correlations can be assigned to primitive uncertain quantities in Python. For a pair, use `Quantity.correlate(x, y, rho)`, where `rho` is between -1 and 1. For a set, use `Quantity.correlate_many(values, covariance_matrix)`. The matrix diagonal must match the input uncertainty variances; the inputs must be scalar primitive quantities. The matrix must be symmetric and positive semidefinite.

For legacy conservative bounds, select interval propagation explicitly:

```python
Quantity.set_propagation_mode("interval")
# Restore the default first-order model when desired:
Quantity.set_propagation_mode("standard")
```

The propagation mode is process-wide. Standard mode is first-order and does not represent distribution shape or nonlinear higher-order effects.

4 Special Units and Conformance Options

Most PUEL units are linear: they convert through a scale factor and can be freely combined by unit algebra. Temperature and logarithmic units are different. They are valid PUEL symbols, but conversion requires specialized semantics.

4.1 Temperature units

`K`, `Cel`, `degF`, and `degR` represent absolute temperature scales. A conversion may contain both a scale and an offset:

```
23*Cel = 296.15*K
```

Consequently, `Cel` is not a multiplicative synonym for `K`. When a temperature factor occurs with one ordinary linear factor, PUQ can apply its specialized conversion to expressions such as `Cel/m`. Rebase-to-base-dimensions is not defined for expressions containing temperature units.

4.2 Logarithmic units

Logarithmic ratios and levels include `AR`, `PR`, `B`, `dB`, `Np`, `dBm`, `dBW`, and the reference levels listed in the unit-system tables. A ratio has no absolute reference; a level does. For example:

```
0*dBm = 1*mW
0*dBW = 1*W
```

The complete tables define each level's reference quantity. PUQ supports a single logarithmic factor with an optional ordinary linear factor, for example `dBmW/Hz`. It does not define arbitrary algebraic rebasing of logarithmic units; rebase-to-base-dimensions is invalid for expressions containing one.

4.3 Conversion availability

Matching physical dimensions are necessary but not sufficient for a special conversion. The source and target special-unit combination must have a defined conversion rule. Otherwise a converter MUST report that the conversion is unavailable rather than applying a linear scale factor.

For conversions between systems whose dimensional conventions differ, a physical-quantity context may also be required. For example, `I` identifies electric current when converting between `statA` and `A`; energy conversion between `erg` and `J` does not require that context.

4.4 Reference implementation feature matrix

The PUQ reference implementation always provides SI. Its compile-time configuration can enable or disable the following optional feature families:

Feature family	PUEL vocabulary affected
CGS systems	`ESU`, `EMU`, and `GU` units and quantity scales
English systems	`IU` and `US` units
Natural systems	`AU`, `PU`, `SRU`, `GRU`, and `GEO` units and quantity scales
Temperature units	`Cel`, `degF`, and `degR` with offset conversions
Logarithmic units	ratios, bel/neper units, and referenced logarithmic levels
Unicode preprocessing	display-oriented Unicode input aliases; portable PUEL remains ASCII

The SciNumTools3 default configuration enables these feature families. A consumer that exchanges PUEL expressions across implementations SHOULD either use the portable SI linear subset or record the required systems and special unit features with the data.

5 Scaling prefixes

Prefix	Factor	Name
`Y`	`1e24`	yotta
`Z`	`1e21`	zetta
`E`	`1e18`	exa
`P`	`1e15`	peta
`T`	`1e12`	tera
`G`	`1e9`	giga
`M`	`1e6`	mega
`k`	`1e3`	kilo
`h`	`1e2`	hecto
`da`	`1e1`	deka
`d`	`1e-1`	deci
`c`	`1e-2`	centi
`m`	`1e-3`	milli
`u`	`1e-6`	micro
`n`	`1e-9`	nano
`p`	`1e-12`	pico
`f`	`1e-15`	femto
`a`	`1e-18`	atto
`z`	`1e-21`	zepto
`y`	`1e-24`	yocto

[!NOTE] The prefix `u` is used to represent micro (Unicode U+03BC) in ASCII-only unit expressions.

6 Unit systems

The following tables specify the built-in vocabulary provided by PUQ for PUEL expressions. A system identifier appears only at the beginning of an expression and is immediately followed by `_`, for example `SI_2*J` or `US_2*gal`. An identifier selects the entire expression; it cannot qualify an individual factor.

Identifier	System	Unit and quantity tables
`SI`	International System of Units	SI definitions
`ESU`	Electrostatic CGS units	ESU definitions
`EMU`	Electromagnetic CGS units	EMU definitions
`GU`	Gaussian CGS units	GU definitions
`IU`	Imperial units	IU definitions
`US`	United States customary units	US definitions
`AU`	Atomic units	AU definitions
`PU`	Planck units	PU definitions
`SRU`	Units for special relativity	SRU definitions
`GRU`	Units for general relativity	GRU definitions
`GEO`	Geometrized units	GEO definitions

6.1 Reading the tables

- Every system includes the shared base-unit symbols. Each system page lists its additional units and aliases, constants, and quantity scales. A unit available in SI is not automatically available in another system.
- Definitions describe one unprefixed unit in terms of symbols from the same system. Prefix permissions apply to the symbol in that row; prefixes are not implicitly inherited through an alias.
- Numerical factors and constant values reproduce the PUQ definitions, including their stated precision. These tables describe the PUEL vocabulary and do not redefine external metrological standards.
- Temperature and logarithmic rows give reference units, not linear equalities. Use the special conversion rules; for example, `Cel` is not equal to `K`.
- SI is always available in PUQ. Other system families, offset temperature units, and logarithmic units depend on enabled implementation features.
- Natural systems often use quantity scales such as `<E>` rather than a new named unit. Constants without `#` may be normalized to unity; fixed `{#...}` reference constants retain their reference values.

7 Shared unit definitions

These symbols are available in every listed system. Prefixes are case-sensitive: **All** means any prefix in the prefix table, **None** means no prefix, and an explicit list permits only those prefixes. An unprefixed symbol is always permitted.

7.1 Base-unit symbols

Symbol	Meaning	Definition or reference	Prefixes
`m`	meter	`m`	All
`g`	gram	`g`	All
`s`	second	`s`	All
`K`	Kelvin	`K`	All
`A`	Ampere	`A`	All
`cd`	candela	`cd`	All
`mol`	mole	`mol`	All
`rad`	radian	`rad`	`m`

7.2 Fixed reference constants

The following constants are available in `SI`, `AU`, `PU`, `SRU`, `GRU`, and `GEO`. The `#` inside the braces distinguishes a fixed reference constant from a system-specific constant: for example, `{#c}` retains its reference value while `{c}` may be `1` in a natural system. Parenthesized digits denote uncertainty using the ordinary PUEL notation.

Symbol	Meaning	Definition or reference	Prefixes
{#alpha}	fine str. const.	7.2973525643(11)e-3	None
{#euler}	Euler's num.	2.718281828459	None
{#N_0}	Avogadro's num.	6.02214076e23	None
{#pi}	pi num.	3.1415926535	None
{#a_0}	Bohr radius	5.29177210544(82)e-11*m	None
{#c}	speed of light	299792458*m/s	None
{#e}	elem. charge	1.602176634e-19*A*s	None
{#eps_0}	permit. of vac.	8.8541878188(14)e-12*m-3*kg-1*s4*A2	None
{#G}	grav. const.	6.67430(15)e-11*m3*kg-1*s-2	None
{#g}	grav. accel.	9.80665*m/s2	None
{#h}	Planck const.	6.62607015e-34*m2*kg*s-1	None
{#hbar}	Reduced Pl. con.	1.054571817e-34*m2*kg*s-1	None
{#k_B}	Boltzmann const.	1.380649e-23*m2*kg*s-2*K-1	None
{#mu_0}	permeab. of vac.	1.25663706127(20)e-6*m*kg*s-2*A-2	None
{#mu_B}	Bohr magneton	9.2740100657(29)e-24*m2*A	None
{#m_e}	electron mass	9.1093837139(28)e-31*kg	None
{#m_p}	proton mass	1.67262192595(52)e-27*kg	None
{#m_n}	neutron mass	1.67492749804(95)e-27*kg	None
{#R_inf}	Rydberg constant	10973731.568157(12)*m-1	None
{#sigma}	Stef.Bolt. const.	5.670374419e-8*kg*s-3*K-4	None
{#N_A}	Avogadro's const.	6.02214076e23/mol	None
{#kap_0}	permittivity	1.11265005620e-13*m-3*g-1*s4*A2	None
{#k_e}	Coulomb const.	8.9875517923(14)e9*m3*kg*s-4*A-2	None
{#H_0}	Hubble const.	67.36(54)*3.24076e-20*s-1	None
{#L_sol}	Solar luminosity	3.828e26*m2*kg*s-3	None
{#M_sol}	Solar mass	1.988475(92)e30*kg	None
{#R_sol}	Solar radius	6.9566(14)e8*m	None

8 SI: International System of Units

Select this system with the leading SI_ prefix. The shared base-unit symbols are also available. Definitions below are interpreted within this system; aliases and capitalization are significant. Prefix rules follow the shared table legend.

The fixed reference constants are also available.

8.1 Derived units and aliases

Symbol	Meaning	Definition or reference	Prefixes
au	astr. unit	149597.870691*Mm	None
AU	astr. unit	au	None
ly	light-year	{c}*yr_j	k, M, G
pc	parsec	3.0857e16*m	k, M, G, T
Ao	Angstrom	1e-10*m	m, k
u	atomic mass unit	g/{N_0}	None
amu	atomic mass unit	u	None
Da	Dalton	u	None
t	tonne	1e3*kg	k, m, G
ct	carat	200*mg	None
min	minute	60*s	None
h	hour	60*min	None
day	day	24*h	None
yr_t	tropical year	365.24219*day	k, m, G
yr_j	Julian year	365.25*day	k, m, G
yr_g	Gregorian year	365.2425*day	k, m, G
yr	year	yr_j	k, m, G
degR	degree Rankine	5/9*K	None

Symbol	Meaning	Definition or reference	Prefixes
deg	angle degree	2*{pi}*rad/360	None
'	angle minute	deg/60	None
''	angle second	'/60	None
ar	are	100*m2	c, d, da, h
b	barn	1e-28*m2	M, k, m, u, n, p, f, a, z, y
l	liter	dm3	All
L	liter	l	All
J	Joule	N*m	All
eV	electronvolt	{e}*V	All
erg	erg	dyn*cm	None
cal	calorie	4.184*J	k, M
Cal	Calorie	kcal	None
Ha	Hartree	4.3597447222071e-18*J	k, M
E_h	Hartree	Ha	k, M
Pa	Pascal	N/m2	All
atm	atm. pressure	101325*Pa	None
bar	bar	100*kPa	M, k, d, c, m
Ba	Barye	0.1*Pa	None
N	Newton	kg*m/s2	All
dyn	dyne	g*cm/s2	All
Gy	Gray	J/kg	None
Sv	Sivert	J/kg	None
T	Tesla	Wb/m2	All
G	Gauss	1e-4*T	All
Hz	Hertz	s-1	All
Bq	Becquerel	s-1	None
W	Watt	J/s	All
hp	horse power	745.7*W	None
kph	kilometer p. hour	km/h	None
kn	knot	0.514444*m/s	None
Mx	Maxwell	1e-8*Wb	None
Wb	Weber	V*s	None
P	Poise	g/(cm*s)	c
St	Stokes	cm2/s	c
Ka	Kayser	cm-1	None
D	Debye	3.33564e-30*C*m	All
Oe	Oersted	1e3*A/(4*{pi}*m)	None
Gal	Gal	cm/s2	None
Bi	Biot	10*A	None
Rad	radiation dose	0.01*Gy	None
sr	steradian	rad2	None
lm	lumen	cd*sr	None
C	coulomb	A*s	All
H	Henry	Wb/A	All
Ohm	Ohm	V/A	All
V	Volt	J/C	All
lx	lux	lm/m2	None
F	Farad	C/V	All
S	Siemens	Ohm-1	All
kat	katal	mol/s	All
dpt	dioptre	1/m	All
Mo	seismic moment	N*m	All
%	percent	1e-2	None
permille	promile	1e-3	None
ppth	promile	1e-3	None
ppm	parts per million	1e-6	None
ppb	parts per billion	1e-9	None
Cel	degree Celsius	K (temperature reference; offset conversion)	None

Symbol	Meaning	Definition or reference	Prefixes
degF	degree Fahrenheit	K (temperature reference; offset conversion)	None
PR	Power ratio	1	None
AR	Amplitude ratio	1	None
Np	Nepers	1 (logarithmic reference; special conversion)	c, d
B	Bel	1 (logarithmic reference; special conversion)	d
Bm	bel-milliwatt	mW (logarithmic reference; special conversion)	d
BmW	bel-milliwatt	mW (logarithmic reference; special conversion)	d
BW	bel-watt	W (logarithmic reference; special conversion)	d
BV	bel-volt	V (logarithmic reference; special conversion)	d
BuV	bel-microvolt	uV (logarithmic reference; special conversion)	d
BA	bel-amps	A (logarithmic reference; special conversion)	d
BuA	bel-microamps	uA (logarithmic reference; special conversion)	d
BOhm	bel-ohms	Ohm (logarithmic reference; special conversion)	d
BSPL	bel-SPL (Pa)	Pa (logarithmic reference; special conversion)	d
BSIL	bel-SIL (W/m2)	W/m2 (logarithmic reference; special conversion)	d
BSWL	bel-SWL (W)	W (logarithmic reference; special conversion)	d
Mw	mom. magnitude	Mo (logarithmic reference; special conversion)	None

8.2 System-specific constants

Symbol	Meaning	Definition or reference	Prefixes
{alpha}	fine str. const.	{#alpha}	None
{euler}	Euler's num.	{#euler}	None
{N_0}	Avogadro's num.	{#N_0}	None
{pi}	pi num.	{#pi}	None
{a_0}	Bohr radius	{#a_0}	None
{c}	speed of light	{#c}	None
{e}	elem. charge	{#e}	None
{eps_0}	permit. of vac.	{#eps_0}	None
{G}	grav. const.	{#G}	None
{g}	grav. accel.	{#g}	None
{h}	Planck const.	{#h}	None
{hbar}	Reduced Pl. con.	{#hbar}	None
{H_0}	Hubble const.	{#H_0}	None
{k_B}	Boltzmann const.	{#k_B}	None
{k_e}	Coulomb const.	{#k_e}	None
{L_sol}	Solar luminosity	{#L_sol}	None
{M_sol}	Solar mass	{#M_sol}	None
{mu_0}	permeab. of vac.	{#mu_0}	None
{mu_B}	Bohr magneton	{#mu_B}	None

Symbol	Meaning	Definition or reference	Prefixes
{m_e}	electron mass	{#m_e}	None
{m_p}	proton mass	{#m_p}	None
{m_n}	neutron mass	{#m_n}	None
{R_inf}	Rydberg constant	{#R_inf}	None
{R_sol}	Solar radius	{#R_sol}	None
{sigma}	Stef.Bolt. const.	{#sigma}	None
{N_A}	Avogadro's const.	{#N_A}	None

8.3 Quantity symbols and conversion factors

<name> denotes the quantity scale in the third column. Where provided, the last column defines |name|, the factor used by contextual conversion to relate that scale to SI. A dash means no explicit factor is defined. The factor expressions are evaluated using SI metre-kilogram-second scaling, and their numerical results are stored as dimensionless conversion coefficients.

Symbol	Meaning	Quantity scale	Conversion factor
<l>	length	m	—
<m>	mass	g	—
<t>	time	s	—
<T>	temperature	K	—
<I>	current	A	—
<Iv>	luminous intensity	cd	—
<n>	amount of substance	mol	—
<the>	angular displacement	rad	—
<v>	velocity	m/s	—
<a>	acceleration	m/s2	—
<jerk>	jerk	m/s3	—
<snap>	snap	m/s4	—
<crac>	crackle	m/s5	—
<pop>	pop	m/s6	—
<rho>	density	kg/m3	—
<rho_A>	area density	kg/m2	—
<rho_l>	linear density	kg/m1	—
<rho_E>	energy density	J/m3	—
<A>	area	m2	—
<V>	volume	m3	—
<p_mom>	momentum	kg*m/s	—
<I_mom>	moment of inertia	kg*m2	—
<ome>	angular velocity	rad/s	—
<a_ome>	angular acceleration	rad/s2	—
<L_ome>	angular momentum	kg*m2/s	—
<F>	force	N	—
<freq>	frequecy	Hz	—
<J_i>	impulse	N*s	—
<Abs>	absement	m*s	—
<mu_v>	dynamic viscosity	Pa*s	—
<nu_v>	kinematic viscosity	m2/s	—
<eps_s>	strain	1	—
<sig_s>	stress	Pa	—
<E_Y>	Young's modulus	Pa	—
<k_s>	spring constant	N/m	—
<gam>	surface tension	J/m2	—
<tau>	torque	N*m	—
<E>	energy	J	—
<S>	entropy	J/K	—
<p>	pressure	Pa	—
<mu_c>	chemical potential	J/mol	—
<W>	work	J	—

Symbol	Meaning	Quantity scale	Conversion factor
<P>	power	W	—
<E_sp>	specific energy	J/kg	—
<C_sp>	specific heat capacity	J/(K*kg)	—
<V_sp>	specific volume	m3/kg	—
<Q>	heat	J	—
<C_p>	head capacity	J/K	—
<phi_Q>	heat flux density	W/m2	—
<C_mol>	molar concentration	mol/m3	—
<E_mol>	molar energy	J/mol	—
<c_mol>	molar heat capacity	J/(K*mol)	—
<S_mol>	molar entropy	J/(K*mol)	—
<rr>	reaction rate	mol/(m3*s)	—
<kap>	thermal conductance	W/K	—
<lam_t>	thermal conductivity	W/(m*K)	—
<R_t>	thermal resistance	K/W	—
<R_lam>	thermal resistivity	K*m/W	—
<Q_vol>	volumetric flow rate	m3/s	—
<q>	electric charge	C	—
<rho_q>	electric charge density	C/m3	—
<p_e>	electric dipole moment	C*m	—
<p_qe>	electric quadrupole moment	C*m2	—
<D_e>	electric displacement field	C/m2	—
<E_e>	electric field strength	V/m	—
<V_e>	electric field gradient	V/m2	—
<alp_e>	electric polarizability	C2*m2*J-1	—
<alp_he>	1st hyperpolarizability	C3*m3*J-2	—
<alp_hhe>	2nd hyperpolarizability	C4*m4*J-3	—
<G_e>	electrical conductance	S	—
<sig_e>	electrical conductivity	S/m	—
<phi_e>	electric potential	V	—
<Phi_E>	electric flux	V*m	—
<R>	electrical resistance	Ohm	—
<rho_e>	electrical resistivity	Ohm*m	—
<Mag>	magnetization	A/m	—
<H>	magnetic field strenght	A/m	—
<Phi_M>	magnetic flux	Wb	—
<B>	magnetic flux density	T	—
<mm>	magnetic dipole moment	N*m/T	—
<mu>	permeability	H/m	—
<eps>	permittivity	F/m	—
<C>	capacitance	F	—
<J_c>	current density	A/m2	—
<Z>	impedance	Ohm	—
<L>	inductance	H	—
<Rel>	reluctance	1/H	—
<P_o>	optical power	dpt	—
<lam_w>	wave length	m	—
<k_n>	wave number	1/m	—
<k_v>	wave vector	1/m	—
<E_il>	illuminance	lx	—
<E_ir>	irradiance	W/m2	—
<F_l>	luinous flux	lm	—
<In>	intensity	W/m2	—
<L_r>	radiance	W/(m2*sr)	—
<I_r>	radiant intensity	W/sr	—
<D_a>	absorbed dose rate	Gy/s	—
<H_de>	dose equivalent	Sv	—
<D_r>	radiation dose	Gy	—
<A_r>	radioactivity	Bq	—

Symbol	Meaning	Quantity scale	Conversion factor
<S_a>	action	J/Hz	—
<spin>	spin	kg*m2*s-1	—

9 ESU: Electrostatic CGS units

Select this system with the leading ESU_ prefix. The shared base-unit symbols are also available. Definitions below are interpreted within this system; aliases and capitalization are significant. Prefix rules follow the shared table legend.

9.1 Derived units and aliases

Symbol	Meaning	Definition or reference	Prefixes
Gal	Gal	cm/s2	None
dyn	dyne	g*cm/s2	All
Ba	Barye	dyn/cm2	None
b	Barye	Ba	None
erg	erg	dyn*cm	None
St	Stokes	cm2/s	c
P	Poise	g/(cm*s)	c
Ka	Kayser	cm-1	None
Pma	permeability	1e-4*m2	None
Rayl	Rayl	g/(cm2*s)	None
radd	radiation dose	100*erg/g	None
Fr	franklin	dyn1:2*cm	None
statC	Statcoulomb	Fr	None
esu	elstat. u.o. charge	Fr	None
statA	Statampere	Fr/s	None
statV	Statvolt	erg/Fr	None
statOhm	Statohm	s/cm	None
statF	Statfarad	cm	None
statH	Stathenry	s2/cm	None
statT	Stattesla	cm-3:2*g1:2	None
statWb	Statweber	cm2*g	None

9.2 System-specific constants

Symbol	Meaning	Definition or reference	Prefixes
{m_u}	atom. mass const.	1.660539069e-24*g	None
{a_0}	Bohr radius	5.291772105e-9*cm	None
{k}	Boltzmann const.	1.380649e-16*erg/K	None
{m_e}	electron mass	9.10938371e-28*g	None
{alpha}	fine-str. const.	0.007297352564	None
{G}	gravit. const.	6.6743e-8*dyn*cm2/g2	None
{h}	Planck const.	6.62607015e-27*erg*s	None
{hbar}	reduced Pl. con.	1.054571817e-27*erg*s	None
{c}	speed of light	2.99792458e10*cm/s	None
{pi}	pi num.	3.1415926	None
{e}	elementary charge	4.80320471e-10*Fr	None
{mu_B}	Bohr magneton	2.780278273e-10*statA*cm2	None

9.3 Quantity symbols and conversion factors

<name> denotes the quantity scale in the third column. Where provided, the last column defines |name|, the factor used by contextual conversion to relate that scale to SI. A dash means no explicit factor is defined. The

factor expressions are evaluated using SI metre-kilogram-second scaling, and their numerical results are stored as dimensionless conversion coefficients.

Symbol	Meaning	Quantity scale	Conversion factor
<l>	length	cm	—
<m>	mass	g	—
<t>	time	s	—
<Iv>	luminous intensity	cd	—
<n>	amount of substance	mol	—
<the>	angular displacement	rad	—
<v>	velocity	cm/s	—
<a>	acceleration	Gal	—
<F>	force	dyn	—
<E>	energy	erg	—
<P>	power	erg/s	—
<p>	pressure	Ba	—
<mu_v>	dynamic viscosity	P	—
<nu_v>	kinematic viscosity	St	—
<k_n>	wave number	Ka	—
<q>	electric charge	statC	1e1/({c}*100)
<I>	current	statA	1e1/({c}*100)
<phi_e>	electric potential	statV	1e-8*({c}*100)
<E_e>	electric field strength	statV/cm	1e-6*({c}*100)
<D_e>	electric displacement field	statC/cm2	1e5/(4*{pi}*({c}*100))
<p_e>	electric dipole moment	statC*cm	1e-1/({c}*100)
<Phi_E>	electric flux	statC	1e1/(4*{pi}*({c}*100))
<eps>	permittivity	1	1e11/(4*{pi}*({c}*100)2)
<R>	electrical resistance	statOhm	({c}*100)2/1e9
<rho_e>	electrical resistivity	statOhm*cm	({c}*100)2/1e11
<C>	capacitance	statF	1e9/({c}*100)2
<L>	inductance	statH	({c}*100)2/1e9
<B>	magnetic flux density	statT	1e-4*({c}*100)
<H>	magnetic field strenght	statA/cm	1e3/(4*{pi}*({c}*100))
<mm>	magnetic dipole moment	statA*cm2	1e-3/({c}*100)
<Phi_M>	magnetic flux	statWb	({c}*100)*1e-8
<mu>	permeability	s2/cm2	4*{pi}*({c}*100)2*1e-7

10 EMU: Electromagnetic CGS units

Select this system with the leading EMU_ prefix. The shared base-unit symbols are also available. Definitions below are interpreted within this system; aliases and capitalization are significant. Prefix rules follow the shared table legend.

10.1 Derived units and aliases

Symbol	Meaning	Definition or reference	Prefixes
Gal	Gal	cm/s2	None
dyn	dyne	g*cm/s2	All
Ba	Barye	dyn/cm2	None
b	Barye	Ba	None
erg	erg	dyn*cm	None
St	Stokes	cm2/s	c
P	Poise	g/(cm*s)	c
Ka	Kayser	cm-1	None
Pma	permeability	1e-4*m2	None
Rayl	Rayl	g/(cm2*s)	None
radd	radiation dose	100*erg/g	None
G	Gauss	cm-1:2*g1:2*s-1	None

Symbol	Meaning	Definition or reference	Prefixes
Oe	Oersted	g1:2/(cm1:2*s)	None
Gb	Gilbert	Oe*cm	None
Mx	Maxwell	dyn1:2*cm2	None
Bi	biot	dyn1:2	None
abA	Abampere	Bi	None
abC	Abcoulomb	Bi*s	None
aC	Abcoulomb	abC	None
abV	Abvolt	g1:2*cm3:2*s-2	None
abOhm	Abohm	cm*s	None
abF	Abfarad	cm-1*s2	None
abH	Abhenry	cm	None

10.2 System-specific constants

Symbol	Meaning	Definition or reference	Prefixes
{m_u}	atom. mass const.	1.660539069e-24*g	None
{a_0}	Bohr radius	5.291772105e-9*cm	None
{k}	Boltzmann const.	1.380649e-16*erg/K	None
{m_e}	electron mass	9.10938371e-28*g	None
{alpha}	fine-str. const.	0.007297352564	None
{G}	gravit. const.	6.6743e-8*dyn*cm2/g2	None
{h}	Planck const.	6.62607015e-27*erg*s	None
{hbar}	reduced Pl. con.	1.054571817e-27*erg*s	None
{c}	speed of light	2.99792458e10*cm/s	None
{pi}	pi num.	3.1415926	None
{mu_B}	Bohr magneton	9.274010066e-21*erg/G	None
{e}	elementary charge	1.602176634e-20*abC	None

10.3 Quantity symbols and conversion factors

<name> denotes the quantity scale in the third column. Where provided, the last column defines |name|, the factor used by contextual conversion to relate that scale to SI. A dash means no explicit factor is defined. The factor expressions are evaluated using SI metre-kilogram-second scaling, and their numerical results are stored as dimensionless conversion coefficients.

Symbol	Meaning	Quantity scale	Conversion factor
<l>	length	cm	—
<m>	mass	g	—
<t>	time	s	—
<Iv>	luminous intensity	cd	—
<n>	amount of substance	mol	—
<the>	angular displacement	rad	—
<v>	velocity	cm/s	—
<a>	acceleration	Gal	—
<F>	force	dyn	—
<E>	energy	erg	—
<P>	power	erg/s	—
<p>	pressure	Ba	—
<mu_v>	dynamic viscosity	P	—
<nu_v>	kinematic viscosity	St	—
<k_n>	wave number	Ka	—
<B>	magnetic flux density	G	1e-4
<H>	magnetic field strenght	Oe	1e3/(4*{pi})
<mm>	magnetic dipole moment	erg/G	1e-3
<Phi_M>	magnetic flux	Mx	1e-8
<mu>	permeability	1	4*{pi}*1e-7

Symbol	Meaning	Quantity scale	Conversion factor
<q>	electric charge	abC	1e1
<I>	current	abA	1e1
<phi_e>	electric potential	abV	1e-8
<E_e>	electric field strength	abV/cm	1e-6
<D_e>	electric displacement field	abC/cm2	1e5/(4*{pi})
<p_e>	electric dipole moment	abC*cm	1e-1
<Phi_E>	electric flux	abC	1e1/(4*{pi})
<eps>	permittivity	s2/cm2	1e11/(4*{pi})
<R>	electrical resistance	abOhm	1e-9
<rho_e>	electrical resistivity	abOhm*cm	1e-11
<C>	capacitance	abF	1e9
<L>	inductance	abH	1e-9

11 GU: Gaussian CGS units

Select this system with the leading GU_ prefix. The shared base-unit symbols are also available. Definitions below are interpreted within this system; aliases and capitalization are significant. Prefix rules follow the shared table legend.

11.1 Derived units and aliases

Symbol	Meaning	Definition or reference	Prefixes
Gal	Gal	cm/s2	None
dyn	dyne	g*cm/s2	All
Ba	Barye	dyn/cm2	None
b	Barye	Ba	None
erg	erg	dyn*cm	None
St	Stokes	cm2/s	c
P	Poise	g/(cm*s)	c
Ka	Kayser	cm-1	None
Pma	permeability	1e-4*m2	None
Rayl	Rayl	g/(cm2*s)	None
radd	radiation dose	100*erg/g	None
Fr	franklin	dyn1:2*cm	None
statC	Statcoulomb	Fr	None
esu	elstat. u.o. charge	Fr	None
statA	Statampere	Fr/s	None
statV	Statvolt	erg/Fr	None
statOhm	Statohm	s/cm	None
statF	Statfarad	cm	None
statH	Stathenry	s2/cm	None
G	Gauss	cm-1:2*g1:2*s-1	None
Oe	Oersted	g1:2/(cm1:2*s)	None
Gb	Gilbert	Oe*cm	None
Mx	Maxwell	dyn1:2*cm2	None

11.2 System-specific constants

Symbol	Meaning	Definition or reference	Prefixes
{m_u}	atom. mass const.	1.660539069e-24*g	None
{a_0}	Bohr radius	5.291772105e-9*cm	None
{k}	Boltzmann const.	1.380649e-16*erg/K	None
{m_e}	electron mass	9.10938371e-28*g	None
{alpha}	fine-str. const.	0.007297352564	None
{G}	gravit. const.	6.6743e-8*dyn*cm2/g2	None

Symbol	Meaning	Definition or reference	Prefixes
{h}	Planck const.	6.62607015e-27*erg*s	None
{hbar}	reduced Pl. con.	1.054571817e-27*erg*s	None
{c}	speed of light	2.99792458e10*cm/s	None
{pi}	pi num.	3.1415926	None
{e}	elementary charge	4.80320471e-10*Fr	None
{mu_B}	Bohr magneton	9.274010066e-21*erg/G	None

11.3 Quantity symbols and conversion factors

<name> denotes the quantity scale in the third column. Where provided, the last column defines |name|, the factor used by contextual conversion to relate that scale to SI. A dash means no explicit factor is defined. The factor expressions are evaluated using SI metre-kilogram-second scaling, and their numerical results are stored as dimensionless conversion coefficients.

Symbol	Meaning	Quantity scale	Conversion factor
<l>	length	cm	—
<m>	mass	g	—
<t>	time	s	—
<Iv>	luminous intensity	cd	—
<n>	amount of substance	mol	—
<the>	angular displacement	rad	—
<v>	velocity	cm/s	—
<a>	acceleration	Gal	—
<F>	force	dyn	—
<E>	energy	erg	—
<P>	power	erg/s	—
<p>	pressure	Ba	—
<mu_v>	dynamic viscosity	P	—
<nu_v>	kinematic viscosity	St	—
<k_n>	wave number	Ka	—
<q>	electric charge	statC	1e1/({c}*100)
<I>	current	statA	1e1/({c}*100)
<phi_e>	electric potential	statV	1e-8*({c}*100)
<E_e>	electric field strength	statV/cm	1e-6*({c}*100)
<D_e>	electric displacement field	statC/cm2	1e5/(4*{pi}*({c}*100))
<p_e>	electric dipole moment	statC*cm	1e-1/({c}*100)
<Phi_E>	electric flux	statC	1e1/(4*{pi}*({c}*100))
<eps>	permittivity	1	1e11/(4*{pi}*({c}*100)2)
<R>	electrical resistance	statOhm	({c}*100)2/1e9
<rho_e>	electrical resistivity	statOhm*cm	({c}*100)2/1e11
<C>	capacitance	statF	1e9/({c}*100)2
<L>	inductance	statH	({c}*100)2/1e9
<B>	magnetic flux density	G	1e-4
<H>	magnetic field strenght	Oe	1e3/(4*{pi})
<mm>	magnetic dipole moment	erg/G	1e-3
<Phi_M>	magnetic flux	Mx	1e-8
<mu>	permeability	1	4*{pi}*1e-7

12 IU: Imperial units

Select this system with the leading IU_ prefix. The shared base-unit symbols are also available. Definitions below are interpreted within this system; aliases and capitalization are significant. Prefix rules follow the shared table legend.

12.1 Derived units and aliases

Symbol	Meaning	Definition or reference	Prefixes
twip	twip	17.6388888e-6*m	None
th	thou	25.4e-6*m	None
mil	mil	th	None
in	inch	0.0254*m	None
''	inch	in	None
li	link	0.201168*m	None
ft	foot	0.3048*m	None
'	foot	ft	None
yd	yard	0.9144*m	None
rd	rod/pole/perch	5.0292*m	None
ch	chain	20.1168*m	None
fur	furlong	201.168*m	None
mi	mile	1609.344*m	None
lea	league	4828.032*m	None
le	league	lea	None
ftm	fathom	1.8288*m	None
cb	cable	219.456*m	None
nmi	nautical mile	1852*m	None
NM	nautical mile	nmi	None
ac	acre	4046.8564224*m2	None
gr	grain	64.79891e-3*g	None
dr	drachm/dram	1.771845195*g	None
oz	ounce	28.349523125*g	None
lb	pound	453.59237*g	None
cwt	long hundredweight	50.80234544e3*g	None
ton	long ton	1.0160469088e6*g	None
slug	slug	14.59390294e3*g	None
min	minute	60*s	None
h	hour	60*min	None
day	day	24*h	None
yr_t	tropical year	365.24219*day	k, m, G
yr_j	Julian year	365.25*day	k, m, G
yr_g	Gregorian year	365.2425*day	k, m, G
yr	year	yr_j	k, m, G
degR	degree Rankine	5/9*K	None
mph	miles per hour	mi/h	None
lbf	pound-force	4.448222e3*g*m*s-2	None
Cel	degree Celsius	K (temperature reference; offset conversion)	None
degF	degree Fahrenheit	K (temperature reference; offset conversion)	None
bc	barleycorn	8.4667e-3*m	None
hh	hand	0.1016*m	None
sqrd	sq. rod/po./per.	25.29285264*m2	None
ro	rood	1011.7141056*m2	None
sqmi	square mile	2589988.110336*m2	None
minim	minim	59.1938802083e-9*m3	None
fls	fluid scruple	1.18387760416e-6*m3	None
fldr	fluid drachm	3.5516328125e-6*m3	None
floz	fluid ounce	28.4130625e-6*m3	None
gi	gill	142.0653125e-6*m3	None
pt	pint	568.26125e-6*m3	None
qt	quart	1136.5225e-6*m3	None
gal	gallon	4.54609e-3*m3	None
bblo	oil barrel	0.16365924*m3	None
st	stone	6.35029318e3*g	None
qr	quarter	12.70058636e3*g	None

12.2 Quantity symbols and conversion factors

`<name>` denotes the quantity scale in the third column. Where provided, the last column defines `|name|`, the factor used by contextual conversion to relate that scale to SI. A dash means no explicit factor is defined. The factor expressions are evaluated using SI metre-kilogram-second scaling, and their numerical results are stored as dimensionless conversion coefficients.

Symbol	Meaning	Quantity scale	Conversion factor
`<l>`	length	`ft`	—
`<m>`	mass	`lb`	—
`<t>`	time	`s`	—
`<Iv>`	luminous intensity	`cd`	—
`<n>`	amount of substance	`mol`	—
`<the>`	angular displacement	`rad`	—
`<V>`	volume	`gal`	—
`<A>`	area	`ac`	—
`<v>`	velocity	`mph`	—

13 US: United States customary units

Select this system with the leading `US_` prefix. The shared base-unit symbols are also available. Definitions below are interpreted within this system; aliases and capitalization are significant. Prefix rules follow the shared table legend.

13.1 Derived units and aliases

Symbol	Meaning	Definition or reference	Prefixes
`twip`	twip	`17.6388888e-6*m`	None
`th`	thou	`25.4e-6*m`	None
`mil`	mil	`th`	None
`in`	inch	`0.0254*m`	None
`''`	inch	`in`	None
`li`	link	`0.201168*m`	None
`ft`	foot	`0.3048*m`	None
`'`	foot	`ft`	None
`yd`	yard	`0.9144*m`	None
`rd`	rod/pole/perch	`5.0292*m`	None
`ch`	chain	`20.1168*m`	None
`fur`	furlong	`201.168*m`	None
`mi`	mile	`1609.344*m`	None
`lea`	league	`4828.032*m`	None
`le`	league	`lea`	None
`ftm`	fathom	`1.8288*m`	None
`cb`	cable	`219.456*m`	None
`nmi`	nautical mile	`1852*m`	None
`NM`	nautical mile	`nmi`	None
`ac`	acre	`4046.8564224*m2`	None
`gr`	grain	`64.79891e-3*g`	None
`dr`	drachm/dram	`1.771845195*g`	None
`oz`	ounce	`28.349523125*g`	None
`lb`	pound	`453.59237*g`	None
`cwt`	long hundredweight	`50.80234544e3*g`	None
`ton`	long ton	`1.0160469088e6*g`	None
`slug`	slug	`14.59390294e3*g`	None
`min`	minute	`60*s`	None
`h`	hour	`60*min`	None
`day`	day	`24*h`	None
`yr_t`	tropical year	`365.24219*day`	`k`, `m`, `G`

Symbol	Meaning	Definition or reference	Prefixes
yr_j	Julian year	365.25*day	k, m, G
yr_g	Gregorian year	365.2425*day	k, m, G
yr	year	yr_j	k, m, G
degR	degree Rankine	5/9*K	None
mph	miles per hour	mi/h	None
lbf	pound-force	4.448222e3*g*m*s-2	None
Cel	degree Celsius	K (temperature reference; offset conversion)	None
degF	degree Fahrenheit	K (temperature reference; offset conversion)	None
p	point	352.778e-6*m	None
P	pica	4.233e-3*m	None
sqft	sq. survey foot	0.09290341*m2	None
sqch	square chain	404.68564224*m2	None
section	section	2.589998e6*m2	None
twp	survey township	93.23993e6*m2	None
cuin	cubic inch	16.387064e-6*m3	None
cuft	cubic foot	28.316846592e-3*m3	None
cuyd	cubic yard	0.764554857984*m3	None
acft	acre-foot	1233.482*m3	None
minim	minim	61.611519921875e-9*m3	None
fldr	fluid drachm	3.6966911953125e-6*m3	None
tsp	teaspoon	4.92892159375e-6*m3	None
tbsp	tablespoon	14.78676478125e-6*m3	None
floz	fluid ounce	29.5735295625e-6*m3	None
jig	jig	44.36029434375e-6*m3	None
gi	gill	118.29411825e-6*m3	None
c	cup	236.5882365e-6*m3	None
pt	pint	0.473176473e-3*m3	None
qt	quart	0.946352946e-3*m3	None
pot	pottle	1.892705892e-3*m3	None
gal	gallon	3.785411784e-3*m3	None
bbl	liquid barrel	119.240471196e-3*m3	None
bblo	oil barrel	158.987294928e-3*m3	None
hhd	hogshead	238.480942392e-3*m3	None
ptd	dry pint	0.55061047135749e-3*m3	None
qtd	dry quart	1.1012209427149e-3*m3	None
gald	dry gallon	4.4048837708599e-3*m3	None
pk	peck	8.8097675417199e-3*m3	None
bu	bushel	35.239070166879e-3*m3	None
bbll	liquid barrel	115.62712358400e-3*m3	None
cwts	short hundredwe.	45.359237e3*g	None
tons	short ton	907.18474e3*g	None
dwt	pennyweight	1.55517384*g	None
ozt	troy ounce	31.1034768*g	None
lbt	troy pound	373.241721*g	None

13.2 Quantity symbols and conversion factors

<name> denotes the quantity scale in the third column. Where provided, the last column defines |name|, the factor used by contextual conversion to relate that scale to SI. A dash means no explicit factor is defined. The factor expressions are evaluated using SI metre-kilogram-second scaling, and their numerical results are stored as dimensionless conversion coefficients.

Symbol	Meaning	Quantity scale	Conversion factor
<l>	length	ft	—
<m>	mass	lb	—
<t>	time	s	—

Symbol	Meaning	Quantity scale	Conversion factor
<Iv>	luminous intensity	cd	—
<n>	amount of substance	mol	—
<the>	angular displacement	rad	—
<V>	volume	gal	—
<A>	area	ac	—
<v>	velocity	mph	—

14 AU: Atomic units

Select this system with the leading AU_ prefix. The shared base-unit symbols are also available. Definitions below are interpreted within this system; aliases and capitalization are significant. Prefix rules follow the shared table legend.

The fixed reference constants are also available.

14.1 Derived units and aliases

Symbol	Meaning	Definition or reference	Prefixes
E_h	hartree	{#hbar}2/({#m_e}*{a_0}2)	None

14.2 System-specific constants

Symbol	Meaning	Definition or reference	Prefixes
{hbar}	reduced Pl. con.	1	None
{e}	elementary charge	1	None
{m_e}	electr. rest mass	1	None
{kap_0}	permittivity	1	None
{a_0}	Bohr radius	{#kap_0}*{#hbar}2/({#m_e}*{#e}2)	None
{alpha}	fine-str. const.	1/137	None
{c}	speed of light	{alpha}-1*{a_0}*E_h/{#hbar}	None
{r_e}	class. e. radius	{alpha}2*{a_0}	None
{lbar}	red. Com. wavele.	{alpha}*{a_0}	None
{m_p}	proton mass	1836*{#m_e}	None

14.3 Quantity symbols and conversion factors

<name> denotes the quantity scale in the third column. Where provided, the last column defines |name|, the factor used by contextual conversion to relate that scale to SI. A dash means no explicit factor is defined. The factor expressions are evaluated using SI metre-kilogram-second scaling, and their numerical results are stored as dimensionless conversion coefficients.

Symbol	Meaning	Quantity scale	Conversion factor
<l>	length	{a_0}	—
<m>	mass	{#m_e}	—
<t>	time	{#hbar}/E_h	—
<I>	current	{#e}*E_h/{#hbar}	—
<Iv>	luminous intensity	cd	—
<n>	amount of substance	mol	—
<the>	angular displacement	rad	—
<v>	velocity	{a_0}*E_h/{#hbar}	—
<p_mom>	momentum	{#hbar}/{a_0}	—
<rho_q>	electric charge density	{#e}/{a_0}3	—
<q>	electric charge	{#e}	—
<p_e>	electric dipole moment	{#e}*{a_0}	—

Symbol	Meaning	Quantity scale	Conversion factor
<p_qe>	electric quadrupole moment	{#e}*{a_0}2	—
<phi_e>	electric potential	E_h/{#e}	—
<E_e>	electric field strength	E_h/{#e}*{a_0}	—
<V_e>	electric field gradient	E_h/{#e}*{a_0}2	—
<eps>	permittivity	{#kap_0}	—
<alp_e>	electric polarizability	{#e}2*{a_0}2/E_h	—
<alp_he>	1st hyperpolarizability	{#e}3*{a_0}3/E_h2	—
<alp_hhe>	2nd hyperpolarizability	{#e}4*{a_0}4/E_h3	—
<B>	magnetic flux density	{#hbar}/({#e}*{a_0}2)	—
<mm>	magnetic dipole moment	{#hbar}*{#e}/{#m_e}	—
<E>	energy	E_h	—
<F>	force	E_h/{a_0}	—
<S_a>	action	{#hbar}	—

15 PU: Planck units

Select this system with the leading PU_ prefix. The shared base-unit symbols are also available. Definitions below are interpreted within this system; aliases and capitalization are significant. Prefix rules follow the shared table legend.

The fixed reference constants are also available.

15.1 Derived units and aliases

This system defines its characteristic scales through the quantity symbols below rather than additional named unit symbols.

15.2 System-specific constants

Symbol	Meaning	Definition or reference	Prefixes
{c}	speed of light	1	None
{G}	grav. const.	1	None
{hbar}	reduced Pl. con.	1	None
{k_B}	Boltzmann const.	1	None

15.3 Quantity symbols and conversion factors

<name> denotes the quantity scale in the third column. Where provided, the last column defines |name|, the factor used by contextual conversion to relate that scale to SI. A dash means no explicit factor is defined. The factor expressions are evaluated using SI metre-kilogram-second scaling, and their numerical results are stored as dimensionless conversion coefficients.

Symbol	Meaning	Quantity scale	Conversion factor
<l>	length	({#hbar}*{#G}/{#c}3)1:2	—
<m>	mass	({#hbar}*{#c}/{#G})1:2	—
<t>	time	({#hbar}*{#G}/{#c}5)1:2	—
<T>	temperature	({#hbar}*{#c}5*{k_B}-2/{#G})1:2	—
<A>	area	{#hbar}*{#G}/{#c}3	—
<V>	volume	({#hbar}*{#G}*{#c}-3)3:2	—
<p_mom>	momentum	({#hbar}*{#c}3/{#G})1:2	—
<E>	energy	({#hbar}*{#c}5/{#G})1:2	—
<F>	force	{#c}4/{#G}	—
<rho>	density	{#c}5/{#hbar}*{#G}-2	—
<a>	acceleration	({#c}7/{#hbar}/{#G})1:2	—

16 SRU: Units for special relativity

Select this system with the leading `SRU_` prefix. The shared base-unit symbols are also available. Definitions below are interpreted within this system; aliases and capitalization are significant. Prefix rules follow the shared table legend.

The fixed reference constants are also available.

16.1 Derived units and aliases

This system defines its characteristic scales through the quantity symbols below rather than additional named unit symbols.

16.2 System-specific constants

Symbol	Meaning	Definition or reference	Prefixes
`{c}`	speed of light	`1`	None

16.3 Quantity symbols and conversion factors

`<name>` denotes the quantity scale in the third column. Where provided, the last column defines `|name|`, the factor used by contextual conversion to relate that scale to SI. A dash means no explicit factor is defined. The factor expressions are evaluated using SI metre-kilogram-second scaling, and their numerical results are stored as dimensionless conversion coefficients.

Symbol	Meaning	Quantity scale	Conversion factor
`<l>`	length	`m`	—
`<m>`	mass	`kg`	—
`<t>`	time	`m`	`{#c}-1`
`<E>`	energy	`kg`	`{#c}2`
`<p_mom>`	momentum	`kg`	`{#c}`
`<v>`	velocity	`1`	`{#c}`
`<L_ome>`	angular momentum	`kg*m`	`{#c}`
`<F>`	force	`kg/m`	`{#c}2`
`<a>`	acceleration	`m-1`	`{#c}2`
`<rho_E>`	energy density	`kg/m3`	`{#c}2`

17 GRU: Units for general relativity

Select this system with the leading `GRU_` prefix. The shared base-unit symbols are also available. Definitions below are interpreted within this system; aliases and capitalization are significant. Prefix rules follow the shared table legend.

The fixed reference constants are also available.

17.1 Derived units and aliases

Symbol	Meaning	Definition or reference	Prefixes
`eV`	electronvolt	`1.602176634e-19*m2*kg*s-2`	`M`, `G`, `T`

17.2 System-specific constants

Symbol	Meaning	Definition or reference	Prefixes
`{c}`	speed of light	`1`	None
`{hbar}`	reduced Pl. con.	`1`	None
`{eps_0}`	permit. of vac.	`1`	None

Symbol	Meaning	Definition or reference	Prefixes
{k_B}	Boltzmann const.	1	None
{pi}	pi num.	{#pi}	None
{alpha}	fine-str. const.	{#e}2/(4*{pi}*{#eps_0}*{#hbar}*{#c})	
{e}	elem. charge	(4*{pi}*{alpha})1:2	None

17.3 Quantity symbols and conversion factors

<name> denotes the quantity scale in the third column. Where provided, the last column defines |name|, the factor used by contextual conversion to relate that scale to SI. A dash means no explicit factor is defined. The factor expressions are evaluated using SI metre-kilogram-second scaling, and their numerical results are stored as dimensionless conversion coefficients.

Symbol	Meaning	Quantity scale	Conversion factor
<l>	length	GeV-1	{#hbar}*{#c}
<m>	mass	GeV	{#c}-2
<t>	time	GeV-1	{#hbar}
<E>	energy	GeV	—
<p_mom>	momentum	GeV	{#c}-1
<v>	velocity	1	{#c}
<L_ome>	angular momentum	1	{#hbar}
<A>	area	GeV-2	({#hbar}*{#c})2
<F>	force	GeV2	({#hbar}*{#c})-1
<rho_E>	energy density	GeV4	({#hbar}*{#c})-3
<q>	electric charge	{#e}/{e}	—

18 GEO: Geometrized units

Select this system with the leading GEO_ prefix. The shared base-unit symbols are also available. Definitions below are interpreted within this system; aliases and capitalization are significant. Prefix rules follow the shared table legend.

The fixed reference constants are also available.

18.1 Derived units and aliases

This system defines its characteristic scales through the quantity symbols below rather than additional named unit symbols.

18.2 System-specific constants

Symbol	Meaning	Definition or reference	Prefixes
{c}	speed of light	1	None
{G}	grav. const.	1	None

18.3 Quantity symbols and conversion factors

<name> denotes the quantity scale in the third column. Where provided, the last column defines |name|, the factor used by contextual conversion to relate that scale to SI. A dash means no explicit factor is defined. The factor expressions are evaluated using SI metre-kilogram-second scaling, and their numerical results are stored as dimensionless conversion coefficients.

Symbol	Meaning	Quantity scale	Conversion factor
<l>	length	m	—
<m>	mass	m	{#c}2/{#G}
<t>	time	m	{#c}-1

Symbol	Meaning	Quantity scale	Conversion factor
<E>	energy	m	{#c}4/{#G}
<p_mom>	momentum	m	{#c}3/{#G}
<v>	velocity	1	{#c}
<L_ome>	angular momentum	m2	{#c}3/{#G}
<F>	force	1	{#c}4/{#G}
<a>	acceleration	m-1	{#c}2
<rho_E>	energy density	m-2	{#c}4/{#G}

A Formal Grammar

This appendix gives the grammar for a standalone PUEL expression in extended Backus–Naur form (EBNF). It is the source grammar used by the PUEL specification and is included here directly from `grammar.ebnf`.

In the grammar, quoted strings are literal characters or tokens. A comma separates grammar elements; | means "or"; square brackets mark an optional element; and braces mark zero or more repetitions. Whitespace shown between grammar elements is for readability and is not permitted inside a portable PUEL expression.

The grammar describes syntactic structure. It does not by itself establish that an identifier exists in the selected unit system, that a prefix is valid for a unit, or that an expression satisfies the validity, uncertainty, and special-unit rules in the preceding chapters. In particular, array literals must contain at least one scalar value, and the portable form contains no whitespace. Those constraints and implementation options are specified in Validity and errors, Uncertainty propagation, and Special units and conformance options.

```
expression          = [ unit-system "_" ] , product ;

product             = factor ,
                      { ( "*" | "/" ) , factor } ;

factor              = unit-system-factor
                    | physical-constant
                    | physical-quantity
                    | grouped-expression
                    | value
                    | unit ;

grouped-expression  = "(" , product , ")" ,
                      [ exponent-value ] ;

value               = scalar-value
                    | array ;

scalar-value        = [ sign ] , mantissa ,
                      [ uncertainty ] ,
                      [ scientific-exponent ] ;

mantissa            = integer , [ "." , { digit } ]
                    | "." , digit , { digit } ;

integer             = digit , { digit } ;

array               = "[" , scalar-value ,
                      { "," , scalar-value } , "]" ;

scientific-exponent = ( "e" | "E" ) , [ sign ] , integer ;

uncertainty         = "(" , integer , ")" ;

unit                = identifier ,
                      [ exponent-value ] ;
```

```
exponent-value      = signed-integer
                    | signed-integer ":" integer ;

signed-integer      = [ sign ] ,
                      integer ;

physical-constant   = "{" , [ "#" ] , identifier , "}" ,
                      [ exponent-value ] ;

physical-quantity   = "<" , identifier , ">" ,
                      [ exponent-value ] ;

unit-system-factor  = "|" , identifier , "|" ,
                      [ exponent-value ] ;

unit-system         = identifier ;

identifier          = letter ,
                      { letter | digit | "_" } ;

sign                = "+" | "-" ;

letter              = "A"…"Z" | "a"…"z" ;

digit               = "0"…"9" ;
```