

Enhanced Spreadsheet Tool with Units

Product Specification

Executive Summary

A next-generation spreadsheet application that treats units as first-class data types, enabling dimensional analysis, automatic unit conversion, and type-safe calculations. Values are stored as tuples (number, unit) rather than formatted numbers, allowing operations like `1ft × 1ft = 1sqft` and `$1000 / 10 users = $100/user` to work naturally.

Core Innovation: Units are data, not formatting. This enables intelligent calculations, collaboration across measurement systems, and AI-native integration via MCP protocol.

1. Core Architecture

1.1 Data Model

Cell Structure: Each cell stores:

- **value:** Numeric value
- **unit:** Unit type (simple or compound)
- **display_unit:** Optional display override (null = use sheet preference)
- **formula:** Optional formula expression

Unit Types:

- **Simple units:** USD, meters, seconds
- **Compound units:** mi/hr, USD/user, m²
- **Dimensionless:** Pure numbers without units

Live Recalculation:

- Formulas recalculate when source cell units change
- If A1 changes from 5m to 5ft, dependent formulas update immediately
- Conversion rates trigger recalculation when updated

1.2 Technology Stack

Backend: Rust for calculation engine and MCP server, performance-critical operations, memory-safe handling of large datasets, WASM compilation for web deployment

Frontend Options: Tauri (Rust-native, lightweight), Electron (wider compatibility), Web app (WASM-based)

MCP Integration: Local MCP server exposes spreadsheet via standard protocol, AI tools can read/write/query with full unit awareness, extensible data sources via MCP servers

2. Unit System

2.1 Built-in Unit Library

Length: Metric (m, cm, mm, km), Imperial (in, ft, yd, mi, nautical mi)

Mass: Metric (g, kg, tonne), Imperial (oz, lb, ton)

Time: s, min, hr, day, week, month, year, Quarters (Q1, Q2, Q3, Q4), Fiscal year handling

Currency: USD, EUR, GBP, JPY, etc. (ISO 4217 currency codes)

Temperature: C (Celsius), F (Fahrenheit), K (Kelvin)

Volume: Metric (L, mL), Imperial (gal, qt, pt, cup, fl oz)

Digital Storage: Decimal (B, KB, MB, GB, TB, PB), Binary (KiB, MiB, GiB, TiB, PiB)

Data Rate: bps, Kbps, Mbps, Gbps

Derived Units: Area (m², ft², acres, hectares), Speed (mi/hr, km/hr, m/s), Force, energy, pressure

2.2 Custom Units

User-Defined Units:

- Simple aliases: "bucks" = USD
- Derived units: "sprint" = 2 weeks
- Domain-specific: "widgets", "API calls", "seats", "PCs"

Scope: Workbook-level custom units, Optional export/import of custom unit libraries

3. Unit Conversion System

3.1 Display vs. Underlying Units

Two Separate Concepts:

1. **Underlying unit** (immutable): What the cell actually stores, never changes unless explicitly converted via CONVERT() function, used for all formula calculations
2. **Display preference** (non-destructive): How units are shown visually, sheet-level or workbook-level toggle, does not affect underlying data or calculations

Example:

- Cell A1 contains 100 feet (underlying)
- Sheet display set to Metric
- A1 displays as 30.48 m (visual only)
- Formula =A1 * 2 calculates as 200 feet (underlying)
- Result displays as 60.96 m

3.2 Unit Preference System

Three-Level Cascade:

1. **User-level default:** Set in application preferences (Metric or Imperial), all new workbooks inherit this setting, auto-detect from system locale on first run
2. **Workbook-level setting (MVP scope):** Overrides user default for specific file, controls mixed-unit resolution (e.g., 5m + 10ft on Metric workbook → 8.048m), preserved when workbook is saved/shared

3. **Sheet-level setting (Future):** Individual sheets override workbook setting, not in MVP but architecture supports it

Important: Opening a workbook does NOT convert its data. A workbook saved with Imperial units keeps those units when opened by a Metric-preference user.

3.3 Display Unit Toggle (Non-Destructive)

Ribbon Toggle Button: "Display: Metric ☒ Imperial" - Instant visual conversion, zero data loss, can switch back and forth freely, sheet-level toggle (workbook-level for MVP)

Behavior: Changes only how values are displayed, formulas continue to calculate with underlying units, export uses underlying units, perfect for international collaboration

3.4 CONVERT() Function (Destructive)

```
Permanent Unit Conversion: =CONVERT(A1, "EUR")
```

Behavior:

- Changes the stored unit permanently
- Cannot be undone (beyond Undo history)
- Also locks that cell's display unit as override
- Cell always shows EUR regardless of sheet display toggle

Warning Dialog: Shows before/after preview, "This will permanently change underlying units. Original values cannot be recovered. Continue?", confirmation required

Use Cases: Standardizing imported data, permanent restructuring for new calculation needs, when you genuinely need different stored units

3.5 Dynamic Conversion Rates

Four Rate Modes:

1. **Live Automatic Realtime Refresh:** Continuously updates from API/MCP server, configurable refresh interval, shows last update time, API source selection (ECB, Fed, Yahoo Finance, etc.)
2. **Prompt on Open:** Dialog when workbook opens ("Update USD→EUR rate? Current: 0.92, Latest: 0.94"), user chooses: Update, Keep Current, or Manual Entry
3. **As of Specific Date:** User selects date from calendar, locks to historical rate from that date, perfect for "reproduce Q2 analysis with Q2 rates"
4. **Manual Entry:** User types assumed rate, displays as "Assumption: 0.95", good for forecast scenarios

Settings Hierarchy:

- **Workbook Default:** Choose one mode (Live Auto | Prompt on Open | As of Date), applies to ALL dynamic conversions by default
- **Per-Conversion Overrides:** Any specific conversion pair can be set to Manual Entry, overrides workbook default just for that pair, everything else follows workbook default

4. Unit Operations & Validation

4.1 Compatible Unit Operations

Automatic Conversion: When operating on compatible units, automatically convert to workbook preference unit:

```
=5m + 10ft (on Metric workbook) → 8.048m  
=5m + 10ft (on Imperial workbook) → 26.404ft
```

Silent, predictable behavior - order doesn't matter, workbook setting controls result.

Comparison Operations: `=5m > 10ft` → TRUE (auto-converts to compare: 5m = 16.404ft)

4.2 Incompatible Unit Operations

Soft Warning Approach: When mixing incompatible units (e.g., 5m + 10s):

- Calculation proceeds (uses numeric values)
- Result is dimensionless (no unit)
- Visual warning: Orange cell background, error icon (■) in cell, tooltip on hover

Philosophy: Warn but don't block. Let users work fluidly and catch issues on review.

4.3 Compound Units

Multiplication:

```
=5m * 3m → 15 m²  
=100USD / 5 users → 20 USD/user  
=50 mi / 2 hr → 25 mi/hr
```

Units combine naturally to create compound units.

Division & Cancellation:

```
=100m / 50m → 2 (dimensionless, units cancel)
=100m / 50s → 2 m/s
```

When units perfectly cancel, result is dimensionless (shows as plain number).

4.4 Edge Cases

Fractional Exponents:

```
=(5m)^2 → 25 m2 (standard, no warning)
=(5m)^0.5 → 2.236 m0.5 (unusual unit)
```

Behavior: Orange warning + error icon, tooltip explaining unusual unit, calculation proceeds (mathematically valid), user decides if valid for their domain

Plain Numbers in Mixed Operations:

```
=5m + 5 (where 5 has no unit)
```

Behavior: Orange warning + error icon, tooltip warns about mixing dimensioned/dimensionless, result is dimensionless (loses unit info)

5. Functions & Formulas

5.1 Unit-Aware Functions

Aggregation Functions (SUM, AVERAGE, MIN, MAX, MEDIAN):

- All inputs auto-convert to workbook preference unit
- Result has that unit

```
=SUM(5m, 10ft, 200cm) (Metric workbook) → 10.048m
```

Statistical Functions (STDEV, VAR, etc.):

- Maintain input unit

```
=STDEV(100 USD, 120 USD, 90 USD) → 15.28 USD
```

Standard deviation of dollars is still in dollars.

COUNT Functions:

- Basic COUNT: =COUNT(5m, 10ft, 3s) → 3 (dimensionless)
- Context-Aware COUNTIF: =COUNTIF(RAM_column, ">16GB") → 6 PCs (inherits row entity unit from table context)

5.2 Mathematical Operations

Arithmetic: Addition/Subtraction auto-convert compatible units, Multiplication/Division create compound units, Exponentiation raises units to power

Trigonometric Functions:

```
=SIN(45°) → 0.7071 (interprets as degrees)  
=SIN(π/4 rad) → 0.7071 (interprets as radians)  
=SIN(45) → 0.8509 (dimensionless assumes radians)
```

All outputs are dimensionless.

Transcendental Functions (LOG, EXP, etc.):

- Accept dimensioned input with warning
- Return dimensionless output

```
=LOG(100m) (orange warning, returns dimensionless)
```

5.3 Text Functions

CONCATENATE: Respects current display settings

Cell A1 = 100 feet (underlying), Sheet display = Metric

```
=CONCATENATE("Length: ", A1) → "Length: 30.48 m"
```

Switch display to Imperial → "Length: 100 ft"

5.4 Unit Conversion Function

CONVERT(value, target_unit):

```
=CONVERT(A1, "EUR")  
=CONVERT(100ft, "m")
```

Behavior: Permanently changes stored unit, locks cell display to target unit, warning dialog before execution, cannot auto-revert

5.5 Literal Units in Formulas

Allowed (edge case): $=5\text{m} * 3\text{m} \rightarrow 15 \text{ m}^2$

Autocomplete works: Type $=5 \text{ m}$ \rightarrow dropdown appears with meters/minutes disambiguation

Use Case: Quick calculations without creating helper cells.

6. Table & Column Metadata

6.1 Column Unit Types

Table Headers with Units:

- Column A header: "Revenue" with unit USD
- Column B header: "Quantity" with unit "items"
- All cells in column default to that unit

Row Entity Units: For tables representing entities (e.g., computer inventory):

- Row unit: "PCs"
- Columns: RAM (GB), Storage (TB), Price (USD)

Context-Aware Operations:

```
=COUNTIF(RAM, ">16GB") → 6 PCs
```

COUNT/COUNTIF inherit row entity unit from table context.

6.2 Named Ranges

Named ranges can have associated units: Range "QuarterlyRevenue" with unit USD, operations on range use that unit context

7. User Interface

7.1 Unit Input

Text Entry with Autocomplete:

1. User types: 100 m
2. Autocomplete dropdown appears showing: m - meters (length), m - minutes (time), mi - miles (length), mm - millimeters (length)
3. Arrow keys to navigate, Tab to accept
4. Category shown in parentheses for disambiguation

Internal vs Display:

- System stores: minutes or meters (unambiguous)
- User sees: m (clean, natural)
- Tooltip on hover can show full name if needed

7.2 Unit Display Styling

Visual Distinction: Cell displaying 100 USD has 100 in normal text, USD in subtle grey box/chip styling (like a tag)

Compound Units: 25 mi/hr → 25 normal, mi/hr in grey box
15 m² → 15 normal, m² in grey box

In Edit Mode: Full text editable, unit portion still visually distinguished, cursor can be anywhere in cell

7.3 Warning Indicators

Visual Warning System:

- **Orange Cell Background:** Incompatible unit operations, fractional exponents, dimensioned + dimensionless mixing
- **Error Icon (■):** Appears in cell with warning, similar to Excel's green triangle
- **Tooltip on Hover:** Explains specific warning

Conversion Rate Indicators:

- ■ Green dot: Live-refreshing rate
- ■ Calendar icon: Fixed date rate
- —■ Pencil icon: Manual assumption
- ■ Yellow warning: Stale/connection issue

7.4 Ribbon & Menus

Display Toggle: Prominent ribbon button "Display: Metric ■ Imperial", one-click switching, shows current mode

Manage Conversion Rates: Ribbon button opens dialog, shows all active conversion pairs, configure modes and manual overrides

Unit Library Browser: Searchable panel, categorized by type (Length, Mass, Currency, etc.), shows recent/favorites

Add Unit to Cell: Right-click menu → "Add Unit", format sidebar with unit dropdown, for adding units to existing numbers

8. Data Import/Export

8.1 Excel Export (MVP - One-Way)

Philosophy: Export is for viewing in Excel, not editing. Excel users should switch to this tool for unit-aware editing.

Export Structure:

- Main Sheets: Side-by-side layout with original columns with values, adjacent columns with unit labels (as text), formulas converted to values or preserved with lookup references
- Metadata Sheets: __units_metadata sheet (mapping cells to units), __conversions sheet (conversion rates used), __README sheet (instructions and warning)

Warning Cell: Cell A1 of first sheet has red/orange formatting: "■■■ EXPORTED FROM [TOOL] - Open in [Tool] for unit-aware editing"

Formula Handling: VLOOKUP/references to metadata sheet for display, maintains some intelligence in Excel, clear it's a viewing format, not editing format

8.2 Native File Format

Serialization: Custom binary or JSON format, full unit metadata preserved, conversion rate history, custom unit definitions, table/column metadata

Backwards Compatibility: Future versions can read older formats

8.3 Copy/Paste Behavior

Within Same Workbook: Units fully preserved, formulas maintain unit awareness

Between Workbooks: Units preserved, custom units copied if not in destination, conversion rates noted if different

Paste into Excel/Other Apps: Paste as "100 USD" (value and unit as text), or offer "Paste Values Only" (strips units)

Paste from Excel: Parse common patterns ("100 USD", "5m", etc.), recognize and convert to unit-aware cells, ambiguous cases prompt user

Paste Special Options: Values only (strip units), units only (preserve structure), convert units on paste

9. MCP Server Integration

9.1 Local MCP Server (Rust)

Exposes Spreadsheet Operations: AI tools can read/write/query with full unit awareness, type-safe unit operations for AI agents, RAG-friendly queries, collaborative human+AI workflows

9.2 MCP Resources

Exposed URIs:

```
spreadsheet://workbook/{id}/sheet/{name}/cell/{ref}
spreadsheet://workbook/{id}/conversions
spreadsheet://workbook/{id}/tables/{name}
spreadsheet://workbook/{id}/units/custom
```

9.3 MCP Tools

Core Operations:

- **read_cell(ref):** Returns cell data with value, unit, formula, display_unit
- **write_cell(ref, value, unit):** Sets cell with unit type
- **query_range(range, filter):** Unit-aware range queries
- **convert_units(value, from_unit, to_unit):** Perform conversion operations
- **get_conversion_rate(pair, date):** Retrieve specific conversion rate
- **search_units(query):** Search unit library

9.4 MCP-Based Conversion Rates

Built-in MCP Servers:

- **mcp-server-currency:** ECB (European Central Bank), Federal Reserve, other FX rate sources
- **mcp-server-stocks:** Yahoo Finance, Alpha Vantage, market data APIs
- **mcp-server-crypto:** Coinbase, CoinGecko, crypto exchange rates

Custom MCP Servers: Enterprise internal rates, proprietary data sources, industry-specific conversions

Fallback Chain: Try MCP Server A, then MCP Server B, then cached value, then manual prompt

Benefits: Extensible data sources, AI-native integration, composable architecture, future-proof design

10. User Experience Principles

10.1 Core Philosophy

Warn, Don't Block: Soft warnings for incompatible operations, let users work fluidly, visual indicators for review, avoid interrupting workflow

Natural Entry: Fast typing (5m, 100USD, 3hr), autocomplete handles ambiguity once, display stays clean and concise, no friction from system complexity

Preserve Intent: Units are core data not formatting, preserved everywhere by default, only stripped when explicitly chosen, conversion is explicit not automatic

International Collaboration: Display toggle for different measurement systems, non-destructive viewing in preferred units, underlying data unchanged, share across regions seamlessly

10.2 Progressive Disclosure

Basic Use: Type numbers with units, formulas work naturally, display toggle for preferences

Advanced Use: Custom units, conversion rate management, table metadata, MCP integration

Power Use: Complex compound units, AI-assisted formula creation, custom MCP servers, advanced unit libraries

11. MVP Scope

11.1 Included in MVP

Core Features: Built-in unit library (SI, Imperial, currency, time, digital storage), custom unit creation, unit-aware formulas and functions, soft warning system, display unit toggle (workbook-level), CONVERT() function with warnings, dynamic conversion rates (4 modes), conversion rate management dialog, table/column unit metadata, context-aware COUNT functions, Excel export (one-way), local MCP server, basic MCP tools/resources, native file format

UI Elements: Text entry with autocomplete, unit display styling, warning indicators (orange cells, tooltips), ribbon toggle for display units, unit library browser, conversion rate dialog

11.2 Future Enhancements

Not in MVP: Sheet-level unit preferences (architecture supports), Excel import (reverse of export), advanced MCP resource types, custom MCP server marketplace, unit version control/history, collaborative editing, mobile apps, plugin system, advanced charting with unit-aware axes, solver with unit constraints

11.3 Technical Debt Allowed in MVP

Simplified conversion rate caching, limited undo history depth, basic error recovery, minimal localization (English + system locale detection), performance optimization deferred for large datasets

12. Success Metrics

12.1 User Experience Metrics

Adoption: Time to first unit-aware formula, percentage of cells using units, custom unit creation rate

Engagement: Display toggle usage frequency, conversion rate dialog interactions, unit library searches

Quality: Warning acknowledgment rate, CONVERT() function usage, copy/paste retention of units

12.2 Technical Metrics

Performance: Formula recalculation time, MCP server response latency, large workbook load time

Reliability: Conversion rate API uptime, data integrity (no lost units), Excel export success rate

AI Integration: MCP tool usage frequency, AI-generated formula accuracy, RAG query success rate

13. Key Differentiators

vs. Traditional Spreadsheets: Units are data not formatting, type-safe operations prevent dimensional errors, automatic unit conversion in formulas, international collaboration without data conversion

vs. Engineering Tools: Familiar spreadsheet interface, business-friendly (currency, time, custom units), AI-native via MCP, easy sharing and export

vs. Programming Libraries: No coding required, visual interface, immediate feedback, accessible to non-programmers

14. Risk Mitigation

14.1 Technical Risks

Performance with Large Datasets: Rust backend for speed, lazy evaluation where possible, caching conversion rates, incremental recalculation

Unit Library Completeness: Start with common domains, custom units fill gaps, community contributions (future), MCP servers for specialized data

Conversion Rate Reliability: Multiple fallback sources, cached values, manual override option, graceful degradation

14.2 UX Risks

Complexity Overwhelm: Progressive disclosure, sane defaults, optional advanced features, good documentation

Learning Curve: Familiar spreadsheet paradigm, autocomplete guides usage, tooltips explain warnings, tutorial workbooks

Adoption Friction: Excel export for compatibility, import common formats, share across teams, clear value proposition

15. Open Questions for Future Resolution

1. Localization strategy for unit names and symbols across languages
2. Versioning approach for custom unit libraries
3. Collaborative editing synchronization of conversion rates
4. Chart/visualization handling of mixed units on axes
5. Solver/optimization with unit constraints
6. Macro/scripting language with unit awareness
7. Database connectivity with unit column types
8. Mobile app UX for unit entry
9. Accessibility for screen readers with units
10. Printing layout with unit display options

Appendix A: Example Use Cases

Financial Analysis: Multi-currency revenue tracking, currency conversion at historical rates, per-user or per-seat pricing calculations, quarterly aggregations with automatic unit handling

Engineering: Mixed imperial/metric conversions, area/volume/force calculations, dimensional analysis verification, unit-aware technical documentation

Data Science: Mixed unit datasets, automatic standardization to SI, time series with various intervals, rate calculations (events/hour, MB/s)

Business Operations: Inventory with multiple measurement systems, international sales tracking, resource allocation (\$/person/month), capacity planning with various units

End of Specification

Version: 1.0 | Last Updated: October 2025 | Status: MVP Definition Complete