A PLAIN-LANGUAGE OVERVIEW

What is TRUGS?

Download this overview as a PDF.

A shared shape for information

People and software can describe the same work in different ways.

For example, “check the result” might mean checking its shape to one person and checking its truth to another.

Such differences make it hard to tell whether people and software mean the same thing.

TRUGS gives them a shared structure that tools can check.

A **graph** is a set of things and their named connections.

A TRUG is one such graph, saved in a structured text format that tools can read.

The format is called JSON: it stores named values as text.

A named value pairs a label with information, such as a person’s name with “Ada”.

Each **node** names a thing and its kind, meaning a category such as file or person.

Each **edge** names a connection between things.

Hierarchy means putting smaller parts inside larger ones.

Validate means check against stated rules.

The shared rules check structure; they do not prove that someone follows an instruction.

A **specification** is the written set of rules a system must follow.

TRUGS stands for Traceable Recursive Universal Graph Specification.

TRUG names one graph; TRUGS names the shared rules.

The name refers to following connections, nesting parts, and sharing rules across subjects.

The TRUGS specification defines the shared structure and language.

Its **vocabulary** is the set of words with defined meanings.

A **branch** adds words for a particular subject, such as calculation.

A branch is a subject-specific extension of the shared rules.

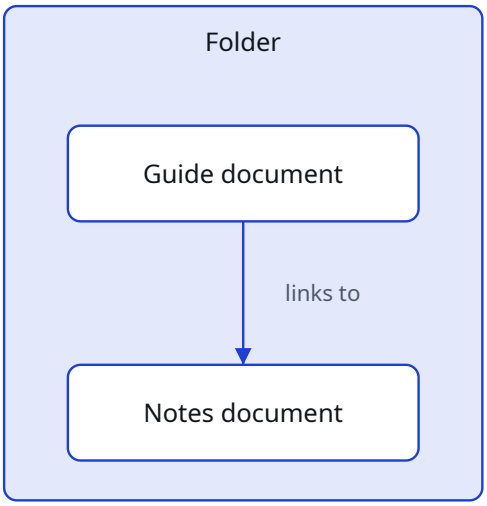
A branch also needs checks for those added meanings.

The four types below use this shared idea for different jobs.

1. Describe — map what is there

Describe maps sources into a graph.

Here, trugs-web reads a folder containing a guide and some notes.



The boxes inside the folder show its contents; the arrow shows that the guide links to the notes.

The nesting of these boxes shows hierarchy.

This is a map of the documents and their link, rather than an answer about whether their contents are true.

2. Instruct — express an instruction

Instruct states what must be done: a sentence in TRL compiles to a graph, which is validated.

TRL is short for TRUG/L, the sentence form of a graph.

The language tool is trugs-tools.

It turns supported sentences into graphs and supported graphs back into sentences.

Supported means the tool recognizes the words and structure used.

Compile means turn a sentence into a graph that tools can check.

When trugs-tools turns a graph back into a sentence, that sentence must compile again.

Otherwise, trugs-tools reports failure.

Formatting, such as spacing, and extra information saved with nodes can be lost during conversion.

For example, a note describing a file may disappear when the graph becomes a sentence.

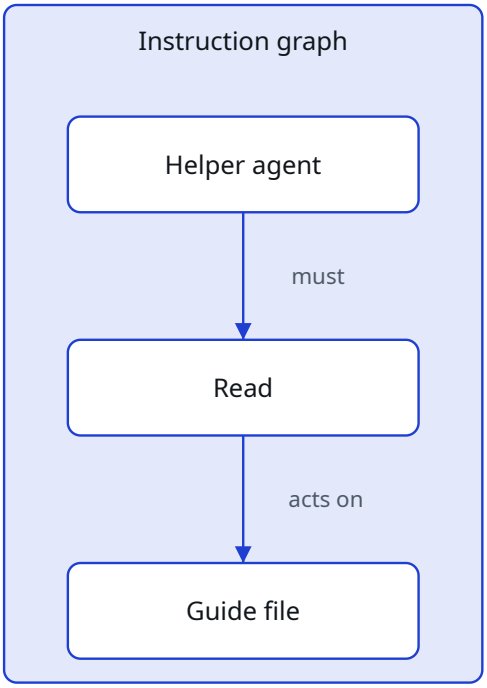
This means conversion need not preserve everything originally saved in a graph.

An agent is a software helper that acts on instructions.

> **AGENT helper SHALL READ FILE guide.**

Here, helper names the agent and guide names the file.

The capital words come from the defined vocabulary; SHALL means must.

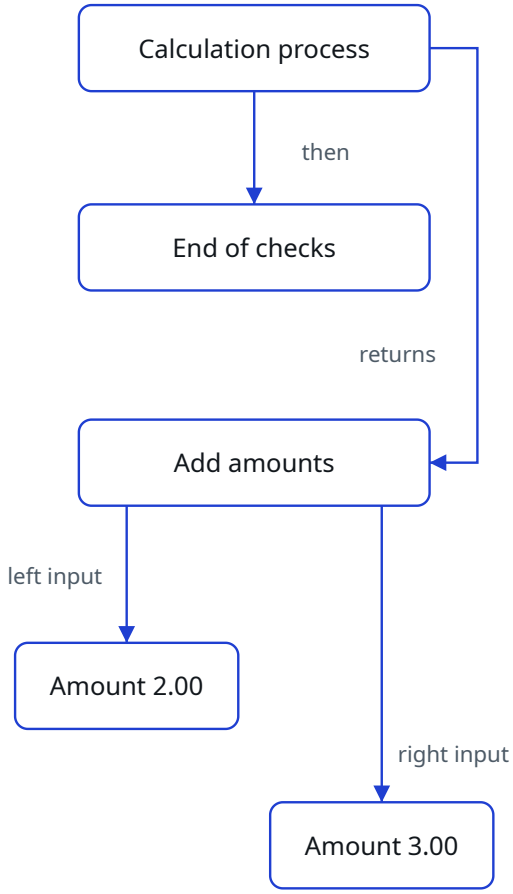


The helper must read the guide.

The outer box holds the instruction; the arrows connect the helper, the reading action and the guide.

3. Compute — represent a calculation

Compute represents calculations as graphs.



The process has no checks to perform; it returns the addition shown below.

“End of checks” marks this empty checking stage; “then” connects the process to that endpoint.

The left and right inputs are the two amounts supplied to the addition.

The addition points to its left amount, 2.00, and its right amount, 3.00.

These numbers are values stored in this graph.

TRUGS-ACCOUNTING-CALCULATOR is a separate tool for the Compute type, with its software shared on GitHub.

It adds calculation words to the shared vocabulary and uses its own compiler, separate from the Instruct language tool.

A compiler is software that translates a description into another form.

The calculator turns calculation graphs into programs: sets of instructions a computer can carry out.

These programs read inputs and return results.

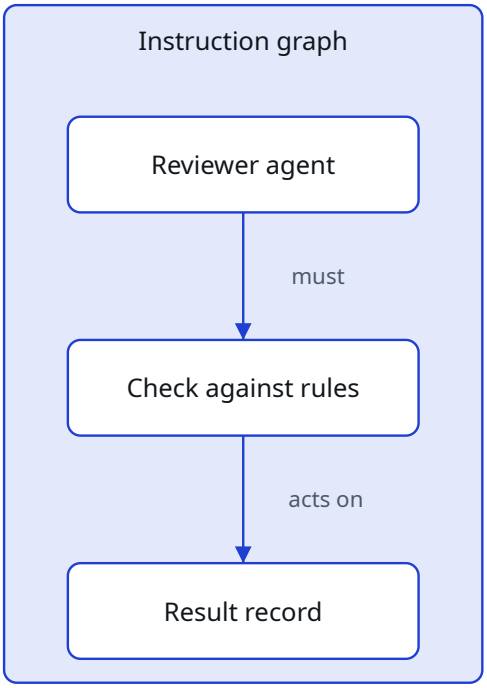
It keeps no accounting records and stores nothing.

Accounting records are a continuing history of money received or spent.

4. Govern — state how work should run

Govern uses graphs to describe how work should run.

A record is a saved piece of information; the result record below holds the result to be checked.



Here, a reviewer must check a result against rules.

The drawing names the reviewer, the checking action and the result.

Structural validation checks a graph's shape against the shared rules.

This graph passes structural validation; that does not show that a review happened.

The Govern runner is TRUGS-RUNNER, with its software shared on GitHub and installable as trugs-runner 0.1.0; it offers and records actions against a fixed graph.

Which part should I use?

A **package** is a bundle of software you can install.

These six parts have different jobs.

The types describe jobs; the six parts provide rules, software and guidance for those jobs.

The specification

Read the specification when you need the rules for the structure and language.

There is nothing to install to read it.

---

trugs-tools

Use trugs-tools to check structure, convert supported language sentences and graphs, or edit graph nodes and connections.

Conversion has the limits described in Instruct above.

---

trugs-store

Use trugs-store when software needs to keep or retrieve graphs.

It supports memory, structured text files and PostgreSQL, a database for keeping organized information.

Here, memory means temporary storage while a program runs; files and databases can keep information after it stops.

It is in beta, meaning a release still being tested.

---

trugs-web

Use trugs-web to map a folder or query a graph.

A query is a request to find information in a graph by following its connections.

For example, a query can find which documents link to a named document.

Query answers follow stored connections; they are not web searches or judgments of truth.

TRUGS-VIEWER

Use TRUGS-VIEWER to open a graph in your browser and explore its things and connections in three dimensions.

Three dimensions means you can turn the view and see depth, as well as height and width.

It is a working demonstration for viewing, with no editing, and is not in active development.

The agent kit

Use the agent kit to give a software helper entry instructions, a short guide and examples.

The kit is called TRUGS-AGENT; its installer is the trugs-agent package.

Entry instructions tell the helper what to read first.

A project is the collection of files for a piece of work.

The installer puts those instructions into that collection.

The kit does not supply a development routine: a sequence for planning, building and reviewing software.

---

trugs-tools and trugs-web both depend on trugs-store, which stores graphs in memory, in JSON files or in PostgreSQL.

For a first practical step, follow the onboarding guide at trugs.ai/onboard/.

What TRUGS is not

TRUGS is not a guarantee that an agent obeys an instruction.

Structural validation is not a judgment that a statement is true.

The released software does not include graphs that rewrite their own structure while running.

Where to read more

GitHub is a website for sharing project material.

Read the specification and examples on GitHub: github.com/TRUGS-LLC/TRUGS.

The paper is “TRUGS: A Formalized Language Specification for Executable Graph Communication Between Humans and Large Language Models”.

Here, “executable” means the graph describes instructions that a computer or software helper can carry out.

It discusses sharing precise instructions between people and software that generates language.

Read the paper’s source on GitHub: github.com/TRUGS-LLC/TRUGS/tree/main/PAPER.

Its published record is on Zenodo, a research archive: doi.org/10.5281/zenodo.19379454.

Companies

One owner holds both TRUGS LLC and Arek Starr LLC. Arek Starr LLC is the only company building TRUGS. TRUGS LLC’s open material needs no Arek Starr LLC product.

Here, open material means the publicly shared specification, tools and guides.

The Companies paragraph explains ownership and whether another company’s product is required.