

TRIZ Skills for Claude

This collection contains 27 TRIZ skills (4 Business-TRIZ, 23 Technical-TRIZ) that can be used as Claude skills — in **Claude Desktop (Cowork)** or **Claude Code**.

Each skill is available as a ready-made ZIP file and can be installed directly.

Installation

In Claude Desktop (Cowork)

1. Open the Claude Desktop app.
2. Go to **Customize** → **Skills**.
3. Click "+" → "+ Create skill".
4. Select the desired ZIP file (e.g., `contradiction-solver.zip`).
5. Done — the skill is immediately available.

Repeat steps 3–4 for each additional skill you want to install.

In Claude Code

1. Unzip the ZIP file.
2. Copy the extracted folder to one of the following locations:

```
# Personal (available across all projects)
cp -r contradiction-solver ~/.claude/skills/

# Or project-specific (available only in this project)
cp -r contradiction-solver .claude/skills/
```

Claude Code will recognize the skill automatically on next launch.

Usage

Automatic Detection

In most cases, you don't need to do anything special. Claude reads the description of each installed skill and decides on its own whether a skill fits the current task. If it does, the skill content is loaded automatically and Claude follows the instructions it contains.

Examples of prompts that automatically trigger the matching skill:

- "Help me solve a technical contradiction in my coffee maker." → `contradiction-solver`
- "How will the banking industry evolve over the next 20 years?" → `business-msd`
- "Analyze the functions of my vacuum cleaner." → `function-analysis`
- "What resources can I make better use of in my production process?" → `resource-analysis`
- "I want to remove components from my system without losing the main function." → `trimming`

Invoking a Specific Skill

If you want to ensure a particular skill is used — regardless of whether Claude would detect it automatically — you can request it explicitly:

In Claude Desktop (Cowork): Use a slash command with the skill name:

```
/contradiction-solver
```

This loads the skill immediately. You can then ask your actual question, or append it directly to the slash command.

In Claude Code: Use a slash command as well:

```
/contradiction-solver
```

Alternative: Mention it in your prompt. You can also ask Claude directly to use a specific skill:

```
Use the "resource-analysis" skill and help me identify resources in my system.
```

or:

```
Please apply the TRIZ 76 Standard Solutions to my problem.
```

As long as you mention the skill name or the method clearly enough, Claude will activate the matching skill.

Structure of a Skill

Each skill consists of a folder with at least one `SKILL.md` file:

```
contradiction-solver/  
├─ SKILL.md           ← Main file (required)  
└─ references/        ← Reference materials (optional)  
    ├─ 40_Inventive_Principles_EN.md  
    ├─ Altshuller_Contradiction_Matrix_AI.csv  
    └─ ...
```

The `SKILL.md` contains:

- **YAML frontmatter** with `name` and `description` — the description determines when Claude activates the skill automatically
- **Markdown instructions** — the step-by-step guide that Claude follows

Some skills additionally contain `references/` (lookup materials such as the contradiction matrix or the 40 inventive principles) and `assets/` (templates such as the ISQ++ questionnaires). These are read by Claude as needed.

Overview of All Skills

Business-TRIZ

Skill	Description
business-msd	Multi-Screen Diagram (9 Boxes) for business and technical systems
perception-mapping	Perception Mapping — surface stakeholder beliefs and contradictions
solutions-at-system-levels	Solution ideas at different system levels (sub-/super-system)
business-function-analysis	Simplified function analysis for non-engineers

Technical-TRIZ

Skill	Description
system-description	Structured system description (Mini-ISQ)
function-analysis	TRIZ function analysis (tool-action-object)
function-analysis-advanced	Advanced function analysis with time/location dimensions
function-interaction-analysis	Component interaction and function analysis
76-standard-solutions	76 Standard Solutions with substance-field analysis
cause-effect-chain-analysis	Cause and Effect Chain Analysis (CECA) and RCA
contradiction-solver	Contradiction solver with 40 inventive principles, Altshuller and 2003 matrix
feature-transfer	Feature transfer between systems
function-oriented-search	Function-oriented search, scientific effects, FOS/MOS
ideality	Increasing ideality of technical systems
innovation-checklist	Innovation Situation Questionnaire (ISQ++)
mpv-analysis	Parameter of Value / Main Parameter of Value analysis
patent-analyzer	Identify contradictions in patent documents
physical-contradictions	Physical contradictions and separation principles
resource-analysis	Resource analysis (6 resource types)
derivative-resources	Derivative resources — discover hidden potential
root-cause-analysis	Root cause analysis for engineering problems

Skill	Description
root-conflict-analysis	Root Conflict Analysis (RCA+) — identify contradiction causes
smart-little-people	Smart Little People (SLP) — creative micro-visualization
system-operator	System Operator (9 Boxes) — evolution and problem solving
trimming	TRIZ trimming with rules A, B, C
interactive-trimming	Interactive trimming with extended rules (A-E, X)
sustainability-assessment	Sustainability assessment with TRIZ function analysis

License

All skills in this collection are released under the **MIT License**. This means you are free to use, copy, modify, and distribute them — including for commercial purposes — as long as the copyright notice and license are retained.

The skills are provided as is, without warranty or liability of any kind.

The full license text is available at [LICENSE](#). The respective authors are noted in each SKILL.md file.