

[New chat](#)[Search chats](#)[Library](#)[Atlas](#)[Projects](#)[GPTs](#)[Explore](#)[Image Reader](#)[Your chats](#)[OpenSCAD parser setup](#)[Code update TopoShape che...](#)[Fix TypeError in Python](#)[Zeeman effect and CO2](#)[Update function check](#)[Git ignore .DS\\_Store](#)[Improved checkObjShape fu...](#)[Hull function in OpenSCAD](#)[UK pensioners full pension d...](#)[Gitignore .DS\\_Store files](#)[FreeCAD API event logging](#)[Rewrite function with excepti...](#)[LibreCAD line selection](#)

Keith Sloan  
Free

[Upgrade](#)

Below is a **clean, practical AST-processing framework** that does exactly what you want:

## ✓ Your requirements restated

You want to:

1. Traverse the AST **starting from highest-level** `hull` and `minkowski` **nodes**.
2. For each such node:
  - Call a function `is_brep_convertible(node)` that determines whether that specific hull/minkowski is *simple enough* to convert to BREP directly inside FreeCAD.
3. **If convertible:**
  - Recursively descend **only through non-hull and non-minkowski children**, processing them normally through your BREP importer.
  - If you encounter another hull/minkowski deeper down:
    - Test again (`is_brep_convertible`).
    - If convertible → process.
    - If NOT convertible → fallback to OpenSCAD (i.e., send the subtree text to OpenSCAD).
4. **If NOT convertible at the top level:**
  - Pass the entire subtree to OpenSCAD as a single request.
5. **All non-hull/minkowski nodes are processed as individual BREP operations.**

This is a **mixed strategy**: pure OpenSCAD only when required, FreeCAD BREP otherwise.

## ✓ Drop-in AST traversal code (works with the AST from previous messages)

`python`[Copy code](#)