

# Parsed Representation

## Overview

The parsed representation is used to convert raw data into objects using the Parsley grammar file. This provides a consistent representation across all parsers without imposing any restrictions on the data. The representation is used to increase readability by splitting the parsing and internal representation into clear steps, making it easier to spot any repetition or inefficient code, and increasing the overall modularity of the program. Anything that will end up in the internal representation (e.g structures and enums) should be converted to the parsed representation before being sent to the internal representation. The internal representation should be modified to accept a ParsedRepresentation object which will contain all the necessary data to convert into a protocol object.

## Boundaries

The parsed representation is simply used to represent data from the parser. It's an intermediate step to ensure that we have all the information before it is type checked and converted into the internal representation. This means that the data in the parsed representation doesn't necessarily need to be correct, as it will be checked when it's converted to the internal representation. Below is a list of things that can be done by the parsed representation:

- Order data
- Make links between data
- Add or remove data

Below is a list of things the parsed representation **cannot** do:

- Enforce naming conventions
- Type checking
- Input validation