

# pre-plantuml

v0.0.1

<https://github.com/SillyFreak/typst-packages/tree/main/pre-plantuml>

**Clemens Koza**

## **ABSTRACT**

Extract PlantUML diagrams from Typst documents to be rendered into images.

## **CONTENTS**

I Introduction .....	2
II Module reference .....	2

# I INTRODUCTION

This package provides two [prequeries](#) for using PlantUML in Typst:

- [plantuml-url\(\)](#) provides the encoded diagram URL to preprocessors for downloading the diagram file;
- [plantuml-source\(\)](#) provides the diagram source code to preprocessors for generating the diagram image locally.

See the Prequery library documentation for more general information on how this can be used.

## II MODULE REFERENCE

### II.a template

- [plantuml-url\(\)](#)
- [plantuml-source\(\)](#)

```
plantuml-url(base-url: string, ..args: arguments, source: raw) -> content
```

A prequery for PlantUML diagrams. Apart from the `base-url` and trailing `source` parameter, the image file name is also mandatory; it is part of `args` for technical reasons. Outside fallback mode, rendering this fails when the rendered diagram file is missing; diagrams need to be rendered in a preprocessing step.

This function provides a dictionary with `url` and `path` as metadata under the label `<web-resource>`. The URL is formed by adding the [encoded form of the diagram](#) to the base URL. This metadata can be queried like this:

```
1 typst query --input prequery-fallback=true --field value ... '<web-resource>'
```

sh

**Fallback:** renders the raw source block.

#### Parameters:

`base-url (string)` – the PlantUML base URL to be used for generated URLs

`..args (arguments)` – arguments to be forwarded to image

`source (raw)` – the PlantUML source code as a raw block

```
plantuml-source(..args: arguments, source: raw) -> content
```

A prequery for PlantUML diagrams. Apart from the trailing `source` parameter, the image file name is also mandatory; it is part of `args` for technical reasons. Outside fallback mode, rendering this fails when the rendered diagram file is missing; diagrams need to be rendered in a preprocessing step.

This function provides a dictionary with `source` and `path` as metadata under the label `<plantuml>`. The sources are strings that can be given to the [PlantUML CLI](#) via stdin. This metadata can be queried like this:

```
1 typst query --input prequery-fallback=true --field value ... '<plantuml>'
```

sh

**Fallback:** renders the raw source block.

**Parameters:**

`..args (arguments)` – arguments to be forwarded to `image`

`source (raw)` – the PlantUML source code as a raw block