# Citus Docker Image

This document describes the steps taken to build a custom Docker image combining **PostgreSQL 16** and **MobilityDB** with **Citus**, with a focus on development compatibility and future integration with extensions like **Distributed MobilityDB**. Unlike the Distributed MobilityDB image, this process required minimal code patching, but still involved manual configuration steps.

The following steps are written under the assumption that you already have a separate Docker image prepared with PostgreSQL 16 and MobilityDB installed. This image serves as the base environment on which additional components or configurations will be applied. For our purposes, we used the Docker image provided in the source code of the MobilityDB extension, with the only modification being that the PostgreSQL version was explicitly set to 16.9. If such an image is not yet available, you will need to build it beforehand, ensuring that both PostgreSQL 16.9 and MobilityDB are properly installed and functional, as the instructions provided here do not cover their installation process.

## Base Image and Goal

* **Base Image:** pg16\_with\_mobilitydb:latest
* **Target Extensions:** Citus

## Key Installation Steps

### 1. Install Build Dependencies

We installed system packages required for building PostgreSQL extensions:

RUN apt-get update && apt-get install -y \  
 git cmake build-essential \  
 clang llvm libzstd-dev libcurl4-openssl-dev \  
 postgresql-server-dev-16 \  
 libpq-dev libproj-dev libgeos-dev libjson-c-dev \  
 libssl-dev pkg-config \  
 liblz4-dev libkrb5-dev \  
 postgresql-16-postgis-3 && \  
 rm -rf /var/lib/apt/lists/\*

### 2. Clone and Build Citus from Source

We cloned the Citus repo without --depth 1 for full source access:

RUN git clone https://github.com/citusdata/citus.git /usr/local/src/citus  
WORKDIR /usr/local/src/citus

Then built Citus with:

RUN ./configure --without-cdc  
**RUN** make -C src/backend/distributed all **&&** make install  
**RUN** make install

This ensured that the CDC module, which is not used in our project, was excluded to simplify dependencies.

## Header File Preparation

### Header Access for MobilityDB Compatibility

To allow future extensions (like DistributedMobilityDB) to compile correctly:

RUN cp -r /usr/local/src/citus/src/include/. /usr/include/postgresql/16/server/

This exposed critical headers such as distributed/pg\_version\_constants.h, distributed/placement.h, and distributed/relation.h to extensions being compiled.

## Summary of Modifications Compared to Vanilla Citus

* Used full Citus source
* Disabled CDC during ./configure
* Copied headers explicitly to PostgreSQL’s server/ include path
* Ensured compatibility with additional spatial/temporal extensions planned for integration

## Result

The final image, tagged e.g. as pg16\_with\_mobility\_citus:latest, is ready for:

* Running PostgreSQL 16 with MobilityDBfull and Citus distributed capabilities
* Serving as a base for extended builds such as Distributed MobilityDB