

# Traceability Pack

In this section:

- [Introduction](#)
- [Requirements](#)
- [Installation and Usage](#)

## Introduction

The Parasoft Traceability Pack is a set of artifacts for your DTP infrastructure that help you establish and demonstrate traceability between tests, code analysis, and peer reviews performed with Parasoft tools and requirements in external application lifecycle management (ALM) systems. The following ALM systems are supported:

- codeBeamer ALM
- Jira - The Xray test management plug-in is also supported
- Polarion ALM
- VersionOne

The Parasoft Traceability Pack also includes the [Jira Issues Widget](#) and a custom flow for sending test data to the connected ALM.

Contact your Parasoft representative for download and licensing information.

The Traceability Pack includes the following components.

## External Application Traceability Reports Flow

This artifact contains the DTP traceability widgets and reports for tracking traceability between the requirements stored in your ALM system and the test, static analysis, and build review data sent to DTP from Parasoft tools (C/C++test, dotTEST, Jtest).

## Sending Test Data to External Application Flow

This artifact automatically calls the `syncTestCases` DTP REST API endpoint when Data Collector receives unit or functional test execution results from Parasoft tools. The API sends the test results from DTP to the associated external application.

## Jira Issues Slice

This artifact adds two widgets that show issues from the connected Jira system. Both widgets are pie charts that organize issues by status or by priority. One widget has settings for querying Jira. The other widget allows you to manually enter a JQL to find issues.

## Requirements

- DTP and DTP Enterprise Pack 2020.1.

## Installation and Usage

Refer to the following chapters for installation and usage details:

- [Integrating with CodeBeamer ALM](#)
- [Integrating with Jira](#)
- [Integrating with Polarion ALM](#)
- [Integrating with VersionOne](#)
- [Jira Issues Widget](#)