Distributed Architecture

Continua CI has been built with a distributed architecture in mind, which grows as your development workload increases. Continua CI consists of one Server which controls and directs the build workflow, and multiple agents that are responsible for building and running actions. By default, Continua CI will install an agent on the Continua Server so right from the start there will always be at least one agent available to Continua CI.

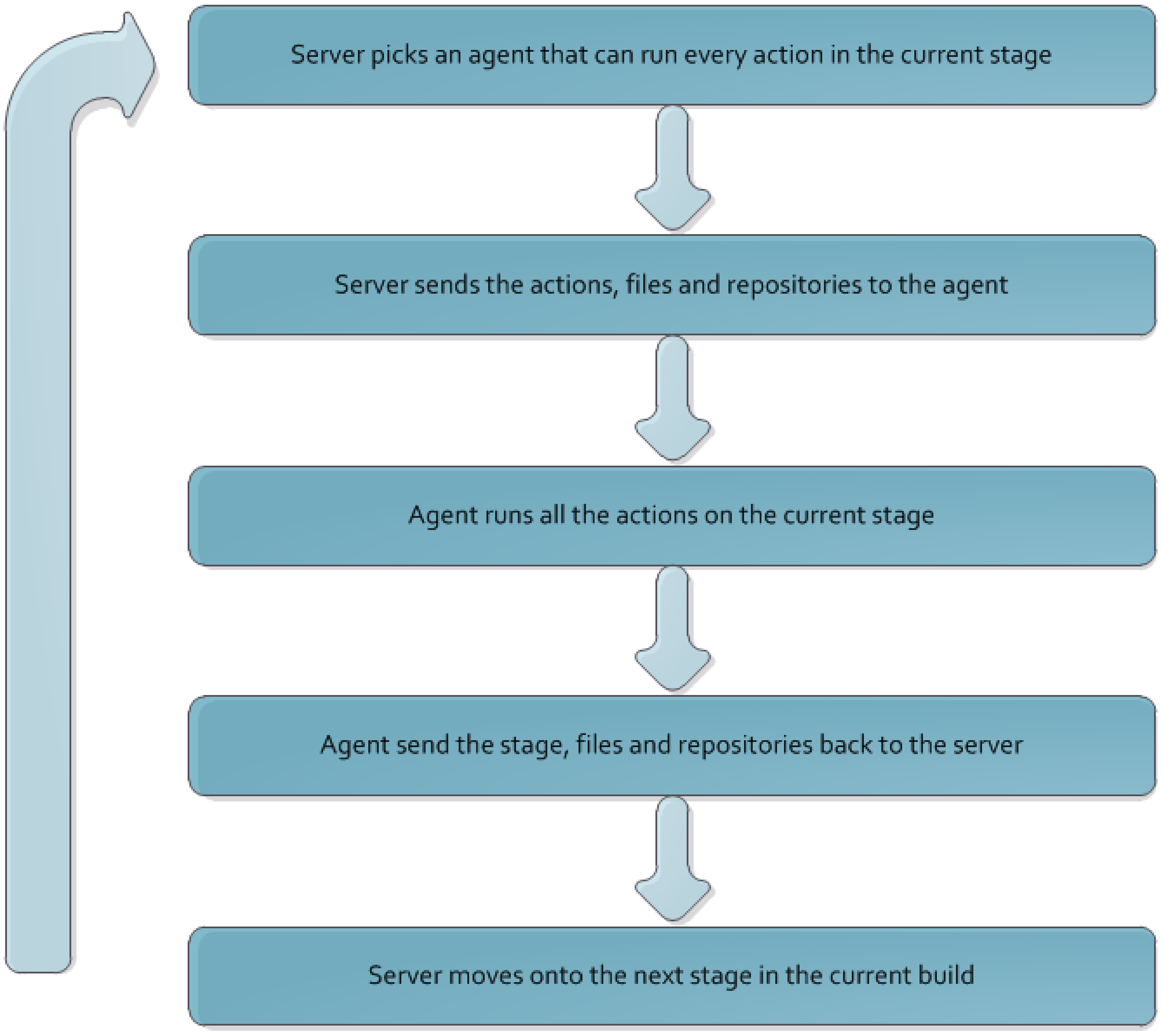
Continua's distributed architecture only comes into effect once extra concurrent builds have been added to your Continua CI license. There is only one version of Continua so no matter which price plan you are currently on, you will always have a fully functioning copy of Continua CI. Where the different price plans differ however is the number of concurrent builds and agents that can be associated with Continua CI. Continua's distributed architecture is only available to users that have installed additional build licenses as Continua's free license only allows users to run one build at a time and this will always be run on the agent installed locally on the Continua server.

Server & Agent Architecture

Continua CI's distributed architecture is based around the idea that the Continua Server will distribute the workload to all the Continua agents. Once a build has begun, the server gets the first stage in the current build and executes the following workflow:

- The server checks all the actions in current stage and picks the next available agent that can run all the actions.
- The server sends all the required information to the agent including repositories, variables, files and actions.
- The agent then steps through and executes each action in the stage.
- The agent sends all the designated information back to the server.
- The server moves onto the next stage and begins the workflow again.

Figure 1. Continua's server and agent architecture


Server picks an agent that can run every action in the current stage
Server sends the actions, files and repositories to the agent
Agent runs all the actions on the current stage
Agent send the stage, files and repositories back to the server
Server moves onto the next stage in the current build

Single Build over Multiple Agents

Continua allows you to break up your build into multiple Stages & Actions. These stages provide a way of splitting up your build into logical sections that can be executed over multiple agents. For example, you could have an agent that is dedicated to FinalBuilder, so any stage that include a FinalBuilder Action will execute on this agent while all other stages will go to the next available agent. This is extremely useful if you have multiple builds that are waiting to execute a FinalBuilder Action, as each build only needs to wait for the stage to finish, rather than the entire build.

Managing Build Files & Controlling the Server to Agent Flow

The information that is sent between the server and an agent is entirely configurable on a stage by stage basis. There are three types of files that are defined at build time:

- **Workspace files:** Workspace files are any files that are used in the Build Workspace. Build workspaces are where builds are executed and generally this is where you put files that the build needs to execute actions. For example, this is where agents temporarily store source code while a project is being built. The build workspace should also be used as the output location for any compiled projects. Workspace Rules define which files should be transferred between the agent and server build workspaces. These rules can be defined on the stages page in the Configuration Wizard. **By default, the Workspace rules are setup so they will only transfer files that are in the Build Workspace's Output subdirectory**. This folder can be accessed with Expressions by using the following syntax: **$Workspace$\Output**.
- **Repository files:** These are files that are pulled from your source repositories and copied into your Build Workspace. Any Source repository files that are required for a particular stage must be transferred to the Build Workspace via Repository Rules. Repository rules are defined per stage and can be set on the stages page in the Configuration Wizard. By default, all files from your source repositories are copied into your build.
- **Artifact files:** These are the files that you want to register as artifacts of the build. Artifact files can be viewed and downloaded directly through the Continua CI interface. Artifact files must be transferred from the agent to the server using Workspace Rules, then they can be registered as artifacts. All artifacts are registered using Artifact Rules which are set on the stages page in the Configuration Wizard. By default, no files are registered as artifacts.