

Getting Started

- [Getting Started with Solr](#)
 - [Downloading Solr](#)
 - [Unpacking Solr](#)
 - [Linux/UNIX](#)
 - [Windows](#)
 - [Understanding Solr startup](#)
 - [Finding cores](#)
 - [Core startup](#)
 - [Running the example](#)

Getting Started with Solr

This page is intended mostly for people who have never used Solr before, but others may find it useful as well. It is not intended for creating a robust production installation, or for [SolrCloud](#). Those topics will be covered on other pages. Here we will work with a binary release. If you want to get the source code instead, this page will not be helpful. See [this page](#) and other resources instead.

Downloading Solr

The [downloads page](#) can help you here. You'll want to get a binary release - either `solr-X.X.X.tgz` or `solr-X.X.X.zip`, depending on what archive tools are available. Solr is a cross-platform Java application, so the contents of both archives are identical, they are just created with different archive tools. Solr releases before 4.1 uses `apache-solr-X.X.X` instead, both as the name of the archive and as the top-level directory found within.

Unpacking Solr

Pick a location on your system where you want to unpack the Solr archive. It is a good idea to make this a short pathname. On Windows, this might be the root of your C: drive. On Linux or one of the many UNIX variants, this might be `/opt`. We are hoping that one day soon there will be installation scripts.

The extracted directory will contain the version number, you might wish to rename this so it is just 'solr' instead. The rest of these instructions will assume that you have chosen the suggested location and that you have renamed the directory to `solr`. It will also refer only to UNIX style pathnames, except in situations that are specific to Windows.

Linux/UNIX

These instructions will leave you with `/opt/solr` as the installation location. You will have to substitute the correct location for `/path/to/downloaded` for this to work. You might also need to use `zcat` instead of `gzcat` for some UNIX variants. If you'd rather use the typical `tar xzf` command, go ahead. The reason that is not done here is because some UNIX variants include a tar program without built-in decompression capability. Note the trailing dash on the command. It is important that you include this.

```
cd /opt
gzcat /path/to/downloaded/solr-4.3.0.tgz | tar xf -
mv solr-4.3.0 solr
cd solr
```

Windows

All recent versions of Windows include support for zipfiles, so download that version of the archive. Double-click on the zipfile and copy the `solr-X.X.X` directory to your chosen location, such as the root of the C: drive. When the copy is done, optionally rename the copy from `solr-X.X.X` to `solr`.

Understanding Solr startup

Before getting into running Solr, it's important to understand the startup process, so that you have an idea what to look for when things go wrong.

The first important thing to know about Solr startup is that Solr itself is not an executable application. It is a servlet, a common way of writing web applications in Java. Servlets run in a servlet container. This point will be expanded in a later section.

Solr pays attention to several Java system properties. One way of setting Java system properties is on the Java commandline, with a syntax of `-Dproperty=value`. Most servlet containers also have ways of setting system properties in their configuration files.

The system property called "solr.solr.home" tells Solr where to begin looking for its configuration. If this property is not defined, Solr uses `./solr` as a default, which means that it looks for a directory named `solr` in the [current working directory](#).

Finding cores

In the `solr.solr.home` location, Solr will look for a `solr.xml` file. This file will give Solr some global configuration information and tell it where it can find its cores. A core is a self-contained Solr index with its own configuration.

Solr4.4 introduces a [new solr.xml format](#) but the [old format](#) will continue to be supported through all 4.x versions.

If `solr.xml` does not exist, Solr may begin looking for a single-core installation. We will not be exploring that installation method here.

Core startup

Once each core is located, either through discovery or the old-style `solr.xml` file, each one will be started if its `loadOnStartup` parameter is not defined or is set to true.

Solr will look for config files in the `conf` subdirectory of the core's `instanceDir`. If the names have not been overridden, this means that Solr will be looking for `conf/solrconfig.xml` and `conf/schema.xml`. The `solrconfig.xml` file may have sections that look for other config files.

Running the example