Writing Simple Salt States

Jeremy Willden
AUTOMATION ENGINEER

@WilldenJeremy   www.constellationlabs.com

Overview

Installing and removing packages

Updating software configuration files

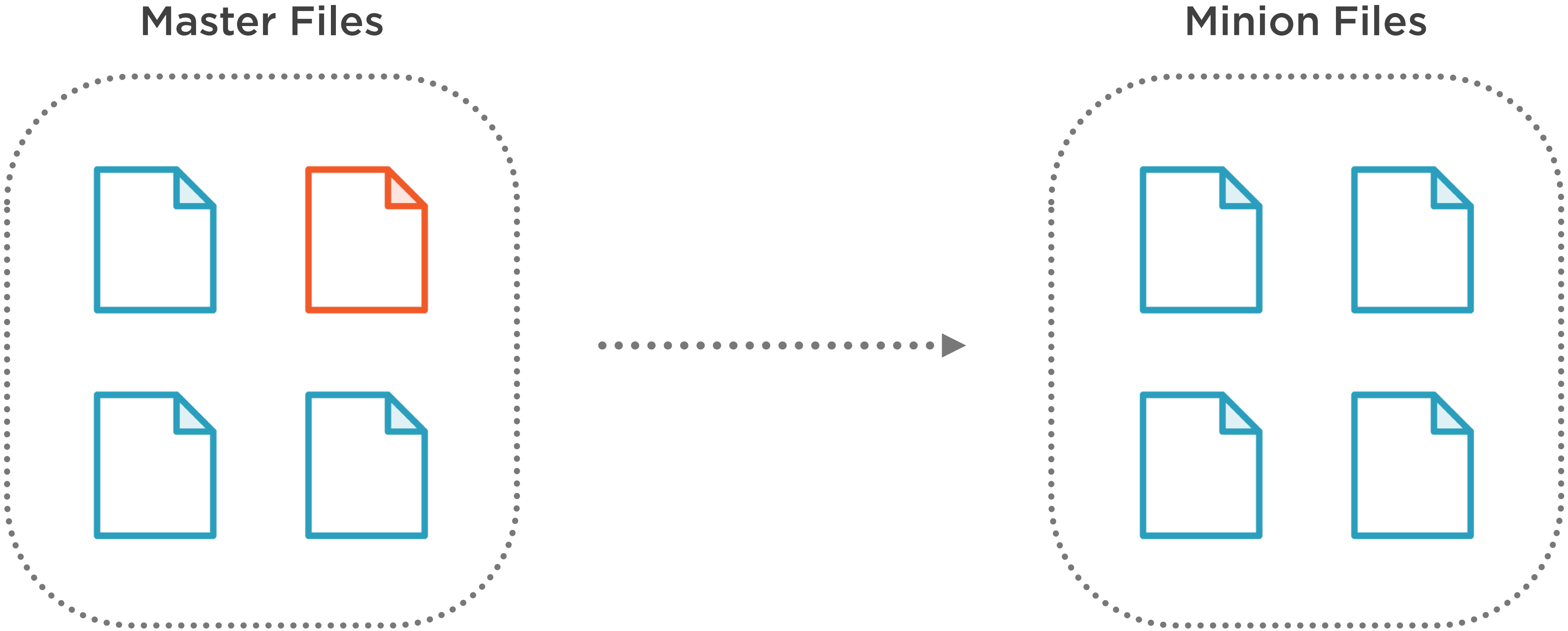
Deploying/synchronizing files

- From the Master
- From a Git repository

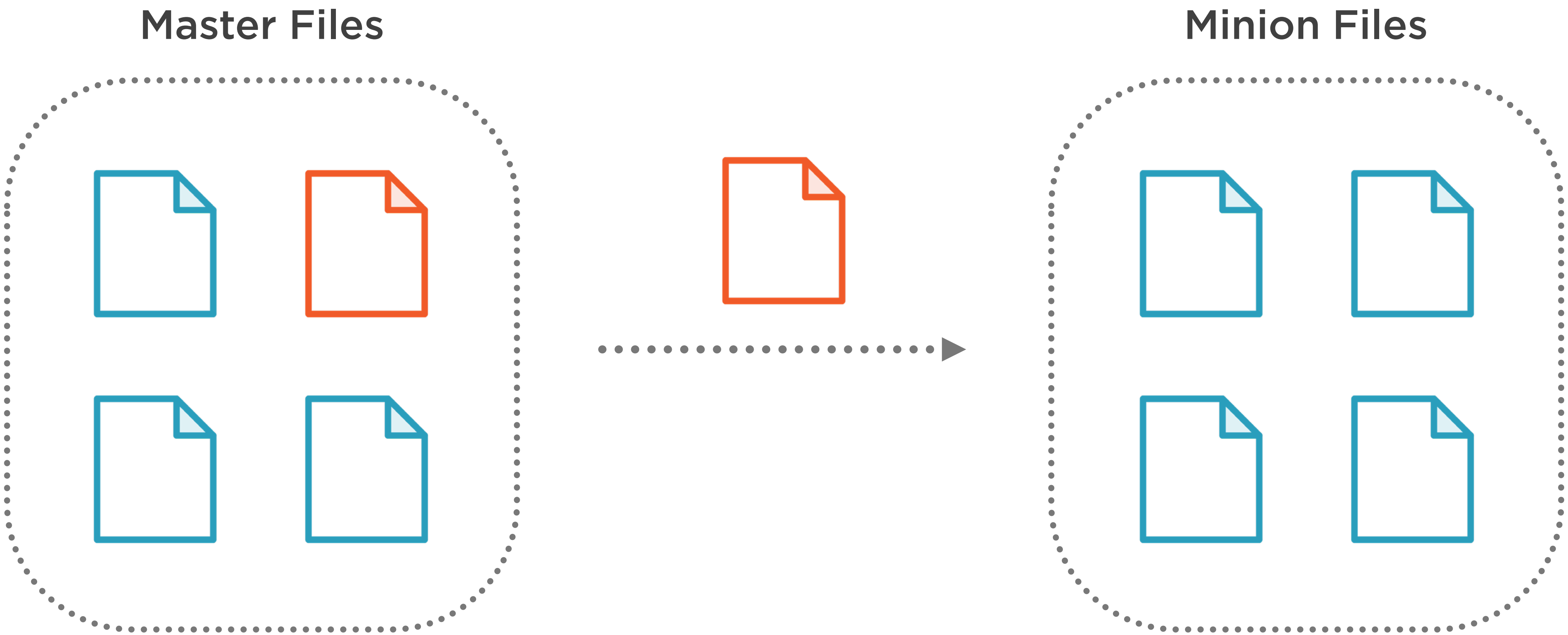
Only updating what is necessary to achieve the correct system state

Talking head

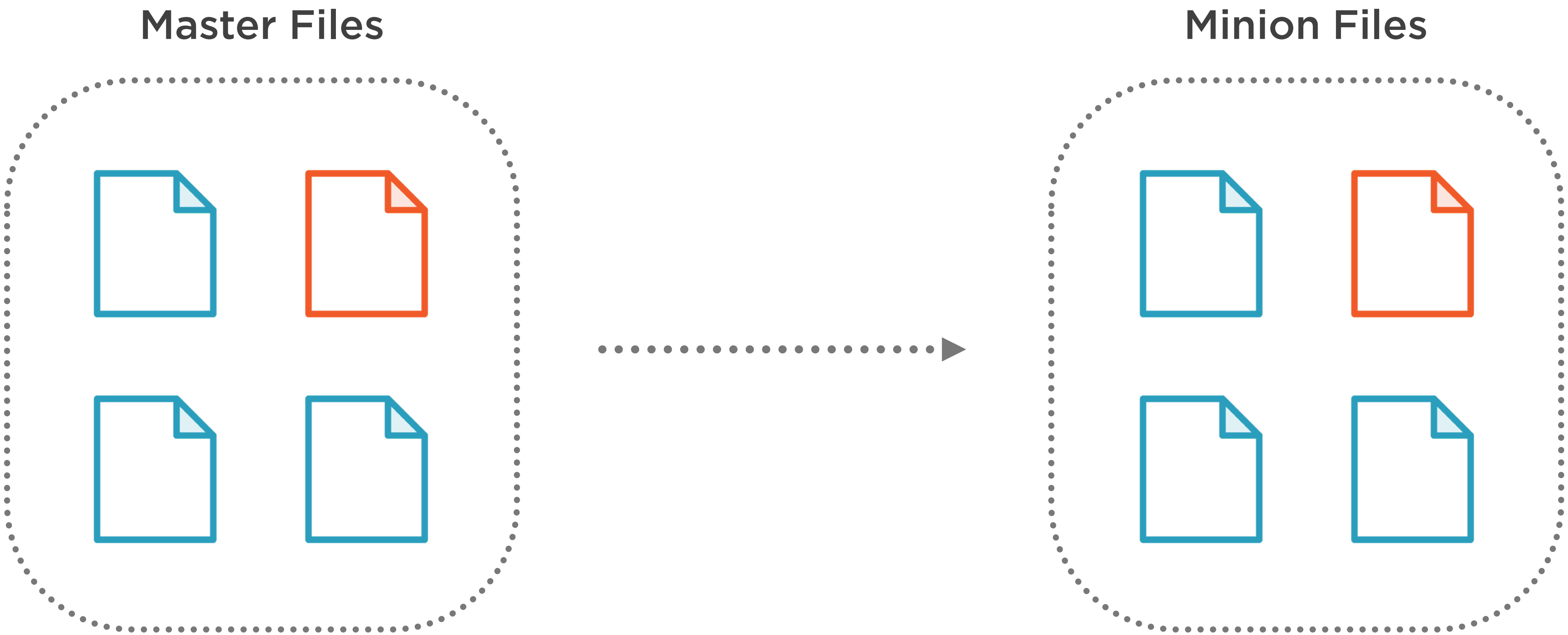
Only Update Necessary Files



Only Update Necessary Files

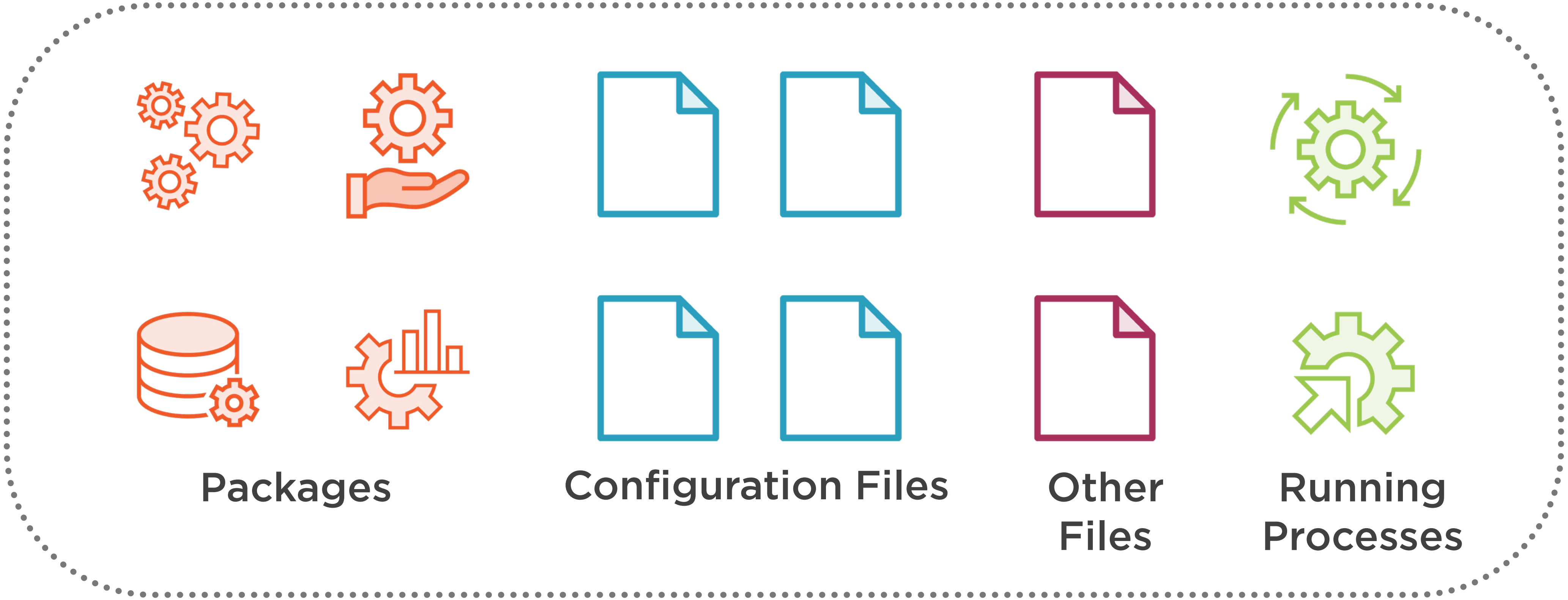


Only Update Necessary Files



States: Packages, Files, and More

Minion State



Packages

Configuration Files

Other Files

Running Processes

State File Organization

```
include:
  - firewall
  - db.routes

Install web server:
  pkg.installed:
    - name: nginx
    - version: ’>=1.14.2’
```

top.sls State File Organization

```
base:
  ‘*’:
    - firewall
    - users.accounts
  ‘dev*’:
    - dev.devtools
  ‘*db’:
    - db.databaseconfig
  ‘*web*’:
    - web.webserver
```

Order Dependency Example

```
include:
  - firewall

httpd:
  pkg.installed: []
  service.running:
    - require:
      - sls: firewall
      - pkg: httpd
```

Demo

Building a basic state file

- Top level file is top.sls

Installing packages

State file hierarchy

Applying the Highstate and single states

Deployment example:

- Web servers
- Database servers
- Development, staging, and production

Demo

Basic packages are installed

Development Machines

- Requested list of tools to install

Add a target to top.sls

- Only applies to development machines

Summary

Salt architecture

- Standard Master-based installation
- Salt-SSH for agent-less operation

Salt Master installation

Salt Minion installation and configuration

Interactive Minion control

- Installing software packages
- Managing files
- Controlling services

Salt States for system management

Preview

Salt formulas

- Pre-built states for common services
- Less effort to build powerful states

What about cloud providers?

- Migration of services
- Abstraction for multiple providers
- Portability to avoid vendor lock-in

Continue learning Salt Open