



Puppet Enterprise with Continuous Delivery: Basic Jumpstart Service Description

Setup, workflows, and pair programming to help your team get fast and sustainable results with Puppet Enterprise and Continuous Delivery for Puppet Enterprise

Overview	3
Recommended for	3
Consulting services description	3
Deliverables	5
Delivery approach	6
Phase 0: Kickoff call, prep, & initial discovery	6
Phase 1: Discovery	6
Phase 2: Installation	6
Phase 3: Workflow	6
Phase 4: Deployment	6
Phase 5: Documentation	7
Timeline	7
Fees	7
Key assumptions	8
Appendix 1 - Engagement technical requirements	9
Appendix 2 - Summary of roles & responsibilities	10

Overview

Puppet Consultants will provide services to help the Customer rapidly deploy Puppet Enterprise with Continuous Delivery and successfully integrate it into their infrastructure. The engagement will consist of knowledge transfer and paired development.

The objective is a working implementation of Puppet Enterprise with Continuous Delivery that is designed to meet the Customer's configuration management automation goals, and a Customer team that is well prepared to continue its configuration management automation effort.

Recommended for

- Installations of up to 500 nodes
- Small teams of up to 3 people
- Customers with operational experience implementing Puppet and Git
- Customers interested in a minimal engagement that focuses on getting a code development and deployment workflow established, without significant customization or new Puppet code
- Self-starters that are comfortable extending configuration management and code deployment workflows on their own, post-engagement.

Consulting services description

Puppet will provide the following services related to the implementation of Puppet Enterprise:

1. Pre-engagement planning and preparation

- a. Discovery call(s)
- b. Draft project plan
- c. Other preparation items as needed (research, testing, etc.)
- d. To be completed by the Customer prior to the start of the engagement:
 - i. Validate appropriate compute, network, and storage has been provisioned.
 - ii. Completion of Continuous Delivery for Puppet Enterprise pre-installation checklist items
 - iii. Confirm full-time availability of at least one technical contact for the duration of the engagement

2. Discovery and concept review

- a. Determine goals and business needs
- b. Provide high-level overview of Puppet concepts, if needed

3. Puppet Enterprise and Continuous Delivery for Puppet Enterprise installation

- a. Installation of Puppet Enterprise in a Monolithic Master configuration

- i. [Monolithic installation documentation](#)
- b. Configure backup and maintenance of the Puppet Enterprise infrastructure
 - i. Knowledge transfer and initial configuration of Puppet Enterprise database backups and maintenance using the https://forge.puppet.com/puppetlabs/pe_database module
 - ii. Knowledge transfer of procedures for restoring the Puppet Enterprise infrastructure from backups
- c. Puppet Enterprise agent deployment
 - i. Knowledge transfer of Puppet Enterprise agent installation
 - ii. Installation of Puppet Enterprise agents on as many nodes as licensed and possible
 - iii. Knowledge transfer of options for automated Puppet Enterprise agent installation using one of the following techniques:
 1. Puppet Frictionless Agent Installation Tool
 2. Puppet Bolt and Tasks for mass agent installation via SSH/WinRM
 3. Windows Group Policy
 4. Suitable pre-existing Customer tooling
 - iv. Knowledge transfer of approaches to integrate Puppet agent installation into the Customer's OS-provisioning system
- d. Assist Customer with initial installation, configuration and usage of Continuous Delivery for Puppet Enterprise
 - i. Integrate Continuous Delivery for Puppet Enterprise with source control and other tools
 - ii. Configure Impact Analysis
 - iii. Add job hardware

4. Puppet module abstraction best practices, as follows

- a. Knowledge transfer on Roles & Profiles pattern, which simplifies the classification of nodes, facilitates code reuse, and makes it simpler for users to have a high-level understanding of what Puppet Enterprise is managing on a system
- b. Pairing with the Customer's staff to create an initial "base" profile to manage common system settings and an additional profile to practice using the pattern
- c. High-level definition of at least 1 additional profile
- d. Knowledge transfer on the use of the Node Classifier, which is used to assign roles, profiles, and component module code to managed nodes
- e. Setup of at least 2 Node Classifier node groups to practice use of the Node Classifier

5. Collaborate on component module development

- a. Pair with the Customer to convert at least 1 short (fewer than 100 lines) scripts into Puppet modules using the version control workflow OR to select appropriate Forge modules as replacements

- 6. Collaborate on Hiera for Code-Data Separation to increase module reusability**
 - a. Pair with the Customer to create a simple Hiera hierarchy containing a demonstrative number of facts (no more than 4 levels) as an introduction to the use of Hiera
 - b. Knowledge transfer on appropriate use of Hiera and effective hierarchy design

- 7. Collaborate on how to use Forge modules**
 - a. Pair with the Customer to select up to 5 appropriate modules from the Puppet Forge for incorporation into the profiles created in step 4(b) to give insight into how to pick modules and assess their quality. Please see Key Assumption 10 for limitation

- 8. Build Continuous Delivery Pipeline and code deployment workflow**
 - a. Pair with Customer to combine steps 4-7 into a single version-controlled repository
 - b. Knowledge transfer on connecting this repository to a Customer provided pre-existing version control repository solution to facilitate collaboration
 - c. Pair with customer to create a continuous delivery pipeline for the control repository

- 9. Deploy Puppet Code with Continuous Delivery for Puppet Enterprise**
 - a. Pair with the Customer to deploy the Profiles, and component modules to one Puppet code environment of the Customer's choice using Continuous Delivery for Puppet Enterprise
 - b. Knowledge transfer on techniques for promoting Puppet code through pre-production environments (i.e. Dev, Test, QA, Staging).

- 10. Recap meeting to close out engagement and to reinforce all the concepts covered**

Deliverables

1. Puppet Enterprise with Continuous Delivery installation
2. Puppet Enterprise agent deployment on as many nodes as possible
3. Code development and deployment workflow setup
4. Pair programming sessions
5. Post engagement documentation including:
 - a. Puppet Enterprise Installation Details
 - b. Workflow Summary
 - c. Recap of Work Performed
 - d. Recommendations

Delivery approach

Our delivery approach is designed to effectively deliver fast time to value with Puppet Enterprise. We work with you throughout the engagement to implement initial base configuration management and continuous delivery and prepare you to extend them to additional nodes and applications.

Phase 0: Kickoff call, prep, & initial discovery

The Puppet Consultant will spend up to 4 hours performing preparation and discovery work prior to the engagement. This includes pre-engagement call(s) with the Customer to discuss logistics and begin the Discovery phase, project planning and preparation, and creation and review of an initial project plan.

Phase 1: Discovery

The Puppet Consultant collaborates with the Customer to understand their technical and business goals, checks their high-level understanding of how Puppet Enterprise works and fills in any gaps, and fills in any blanks in the project plan.

Phase 2: Installation

The Puppet Consultant and Customer install Puppet Enterprise and Continuous Delivery for Puppet Enterprise and configure the various workflow components. Time is spent reviewing the nuts and bolts of maintaining these systems to ensure the Customer understands and can maintain them ongoing.

Phase 3: Workflow

The Puppet Consultant collaborates with the Customer team to design and implement a module development workflow that is tailored to the Customer's organizational structure needs. The Puppet Consultant then pairs with the Customer team to use this workflow to convert an existing script into a Puppet module in order to provide hands-on experience with the workflow. The Customer also builds a continuous delivery pipeline to be used in Puppet code deployment from their source control repository.

Phase 4: Deployment

The Puppet Consultant pairs with the Customer team to bring nodes and services under Puppet Enterprise management. Some base configuration items are managed on at least some test nodes by the time the engagement ends. Continuous Delivery for Puppet Enterprise is fully configured and deploying Customer Puppet code to an environment in their infrastructure. Next steps and recommendations are discussed and the time spent pairing leaves Customer with hands-on experience bringing configuration items under management with Puppet Enterprise and getting experience with a continuous delivery code deployment workflow.



Phase 5: Documentation

The Puppet Consultant will spend four (4) hours completing engagement documentation. The final documentation deliverable will be provided to Customer within five (5) business days of the end of the engagement and will consist of the following:

- Engagement recap
- Architecture description and diagrams
- Workflow description and usage
- Recommendations for future enhancements to the Puppet Enterprise solution

Timeline

Phases 1-4 of the Puppet Enterprise with Continuous Delivery Basic Jumpstart (as outlined in this service description) require 8 days (64 hours), delivered over two weeks to complete. In addition, 16 hours are set aside for preparation and documentation, with each requiring approximately 8 hours. The chart below reflects the expected project timeline.

Phase	Pre-engagement	Week 1		Week 2		Post-engagement
Preparation						
Discovery						
Installation						
Workflow						
Deployment						
Documentation						

Fees

Product description	Hours	Cost
Puppet Enterprise with Continuous Delivery Jumpstart: Basic	Up to 80	\$23,995.00

The fees for this project will be a fixed price for up to a maximum of 80 hours. Puppet will use best efforts to complete the Services within this timeframe, but this project will be deemed completed if the maximum number of hours is reached.

Services for this engagement will be provided remotely, unless Customer and Puppet mutually agree the Services will be performed on-site. In addition to the fees above, Customer will be responsible for travel and expenses for on-site Services. A travel and expense estimate will be provided on the quote and billed at actuals not to exceed the

estimate, unless otherwise specified in the quote that travel and expenses will be invoiced in full at the time of purchase. Any quote that does not include a travel and expense estimate will be performed remotely.

Key assumptions

The following assumptions are reflected in the services, timeline, and estimated effort outlined in this service description:

1. Phases 1-4 of the engagement (as outlined in this service description) will be performed in one week during normal business hours (Monday through Thursday, 8:00 a.m. - 5:00 p.m. local time), unless otherwise agreed to in writing by the parties.
2. Customer staff has attended Getting Started with Puppet training within the last 12 months or has equivalent operational experience with Puppet.
3. Customer will provide prompt feedback on all deliverables.
4. Customer will provide prompt access to all systems and resources that Puppet will need in order to complete the work.
5. Customer must provide a single point of contact that will be available at least 75% of the time throughout the duration of the engagement.
6. Customer will provide Puppet with documentation and access to subject matter experts for non-Puppet systems/software within the scope of the engagement.
7. Customer will have identified key personnel prior to the beginning of the engagement.
8. Customer will have all necessary security exceptions, firewall rules, network routers, computer and storage resources available prior to the start of the engagement, as detailed in Appendix 1 of this service description.
9. Puppet does not provide support for third-party software that is implemented as part of a Puppet Enterprise solution, such as version control systems, repository management, packaging, and other software that is not part of the Puppet Enterprise stack.
10. Forge modules may not exist for the Customer's specific use case. Puppet makes no guarantees about specific modules being available on the Puppet Forge. Any alteration to pre-existing Forge modules to meet Customer needs will be at Puppet's discretion and will be the responsibility of the Customer to support and maintain post-engagement.
11. Module development and/or automation task implementation is confined to work that can be reasonably completed within the engagement's allocated hours. As a result, such work may not occur, may be incomplete, and/or may require further effort from the Customer to complete post-engagement.

Appendix 1 - Engagement technical requirements

A successful consulting engagement requires advance preparation. Failure to meet these pre-engagement requirements will have a direct impact on the completion of all the goals of the engagement.

The Customer is expected to review all requirements. Any requirements that are not met should be reported during the pre-engagement call or earlier. Puppet and the Customer will discuss the issues and determine whether to delay the engagement or attempt to work around the issue during the engagement.

Basic Jumpstart requirements

[The pre-engagement technical requirements](#) must be verified and completed by the Customer prior to the engagement. Failure to complete these will negatively impact Puppet's ability to meet all deliverables in the time allotted. Puppet bears no responsibility for delays due to incomplete pre-engagement requirements.

Appendix 2 - Summary of roles & responsibilities

- Puppet Consultant:** This refers to the Puppet Consultant who will be responsible for the delivery of the engagement. They will be involved in most if not all aspects of the engagement.
- Puppet Practitioner:** The Puppet Practitioner is the Customer’s technical point of contact who is primarily responsible for the Puppet Enterprise implementation and is assumed to be championing the continued expansion of the usage of Puppet Enterprise post-engagement. They bear the majority of the responsibility for working with Puppet during the engagement and are expected to be present for ~75% of the engagement. Their schedule should be adjusted to ensure they are able to give the engagement the focus necessary.
- Manager/Director:** This is most often the buyer or project sponsor at the Customer. Their role is to be aware of what is happening as part of the engagement and act as a point of contact/escalation should any assistance be needed from other groups, or if work is blocked at a level that the Practitioner cannot resolve.
- Subject Matter Experts:** Subject Matter Experts are the technical specialists in non-Puppet systems/software who are aware of but may not be heavily involved in the use of Puppet Enterprise. This may include networking, provisioning, database, application, and development teams, or any other group that might be valuable to speak with as part of this engagement.

Note: Any group that has the potential to block progress during the engagement should be informed that the engagement is taking place so they are equipped to respond to any urgent requests.

Responsibilities (? = may need to be involved)	Puppet Consultant	Puppet Practitioner	Manager/Director	SME
Preparation				
Review and verify pre-engagement requirements are met via email		✓	?	?
Pre-engagement preparation	✓	✓	✓	?
Pre-engagement kickoff call	✓	✓	✓	?
Pre-engagement discovery call(s)	✓	✓		?
Discovery				
Kickoff/discovery meeting	✓	✓	✓	✓
Finalize project plan	✓	✓	✓	

Puppet concept overview (if required)	✓	✓		
Install				
Install Puppet Enterprise in monolithic configuration	✓	✓		
Review Puppet Enterprise maintenance & backups	✓	✓		?
Install Puppet Enterprise agent on as many nodes as possible	✓	✓		
Knowledge transfer on how to automate install of remaining agents	✓	✓		
Pair on automating agent install during provisioning	✓	✓		?
Install and initially configure Continuous Delivery for Puppet Enterprise	✓	✓		
Workflow				
Discuss and set up git server, if required	✓	✓		?
Knowledge transfer of version control workflow	✓	✓		
Pair programming to practice workflow	✓	✓		
Knowledge transfer on roles & profiles	✓	✓		
Knowledge transfer on Hiera & creation of initial hierarchy	✓	✓		?
Build continuous delivery code deployment pipeline	✓	✓		
Deploy				
Incorporate Forge modules (SME needed for details Practitioner may not have)	✓	✓		?
Convert existing script(s) into Puppet modules	✓	✓		
Deploy managed configurations on pre-production environments with Continuous Delivery for Puppet Enterprise	✓	✓		
Handoff call with support	✓	✓	✓	
Close-out meeting	✓	✓	✓	?
Documentation				
Create final documentation	✓			
Review and check-in final documentation to version control		✓	?	

? = May be needed

✓ = Required