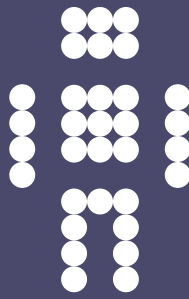


CFEngine



CFEngine Nova Mission Portal

CFEngine Enterprise Documentation
for version 2.1 alpha2, 18. July 2011

CFEngine

Table of Contents

1	Introduction	1
2	Mission Portal	3
2.1	Mission Portal Rooms	3
2.1.1	Mission Status	4
2.1.2	Mission Engineering	5
2.1.3	Mission Planning	6
2.1.4	Mission Library	7
2.2	Finders	7
2.2.1	Host finder	8
2.2.2	Class finder	8
2.2.3	Promise finder	9
2.2.4	Report finder	10
2.2.5	Topic finder	11
2.3	Viewers	11
2.3.1	Host viewer	11
2.3.2	Bundle viewer	12
2.3.3	Promise viewer	13
2.3.4	Vital signs viewer	14
2.3.5	Topics viewer (Knowledge map)	14
2.3.6	Weakest hosts viewer	15
2.3.7	Report viewer	15
2.4	Editors	16
2.4.1	Policy editor	16

1 Introduction

CFEngine Nova is a commercially licensed version of the core CFEngine software¹ with enterprise library extensions. All of the documentation for CFEngine 3 applies to CFEngine Nova. This document is a supplement describing the graphical user interphase used in CFEngine Nova: the Mission Portal.

Knowledge management has become a new focus in IT management, and it is a core focus at CFEngine. Comprehending the growing complexity of IT operations is one of the main challenges in IT today. CFEngine Nova has a number of key features for knowledge management, including automated documentation, report generation and associative inference. The CFEngine Mission Portal is the centerpiece of user interaction with CFEngine Nova and can be accessed after the successful installation of the software on your hub server (policy host). Connect to the server (port 80) with your web browser (for example at <http://123.456.789.123>), you should see a login screen:



Figure: Mission Portal login screen

Default user and password is 'admin' and 'admin' (make sure to change this at first login to prevent unauthorized access).

¹ Major version 3

2 Mission Portal

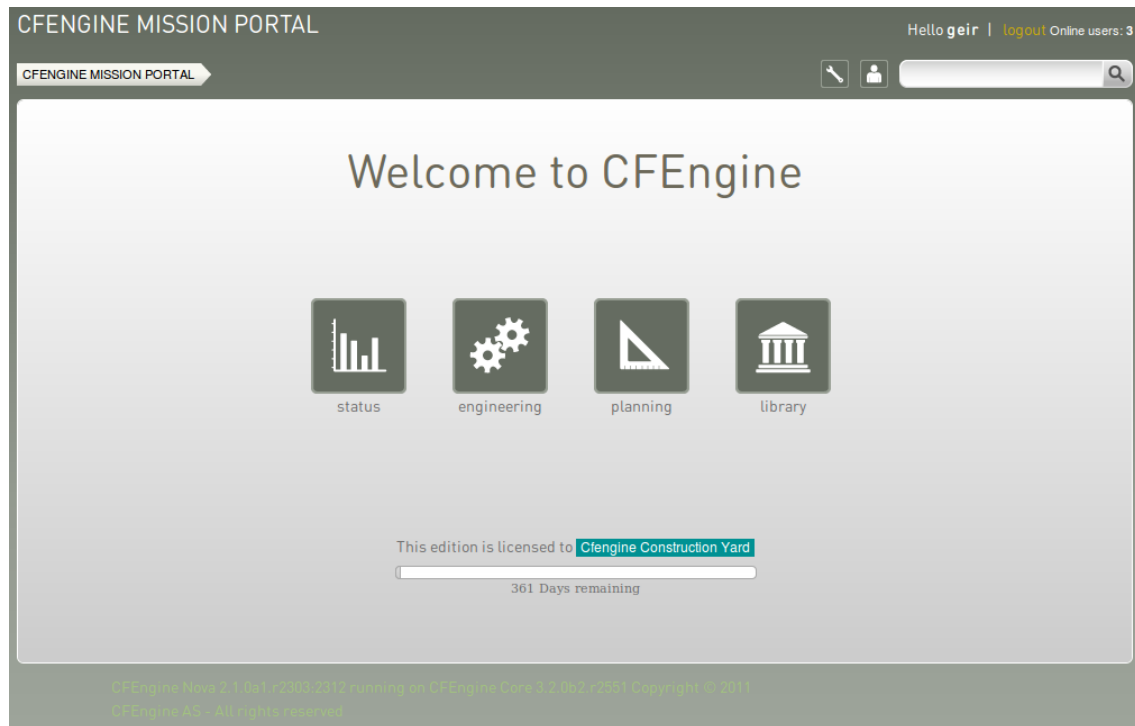


Figure: The mission portal

2.1 Mission Portal Rooms

There are four main rooms in the Mission Portal that offer insight into different aspects of operations:

- Mission status - a top level overview of compliance status and business value
- Mission engineering - a place to see the current state of system repair
- Mission planning - a place to plan and make policy changes
- Mission library - a knowledge bank that connects information together

Each of these rooms is a beginning from which you can refine your overview and search through information.

2.1.1 Mission Status

Mission status is a high level summary of how well the entire system is behaving. Note that the charts will show blank (black) after a fresh install. The Host Status will be populated first, followed by the Business Value and Compliance Summary charts after approximately 6 hours.

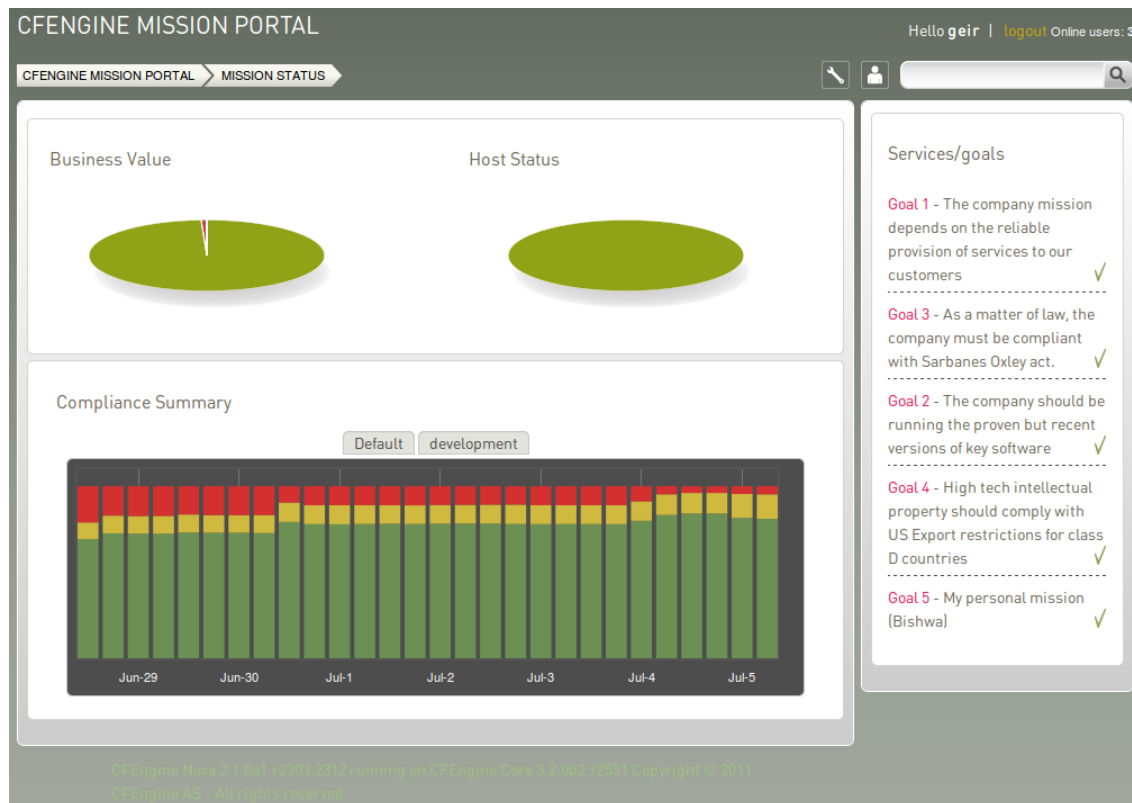


Figure: The status of IT operations.

Business Value and Host Status: Pie charts show the business value of the promises kept/not kept as well as host status. In Host Status, each host represents a slice of the pie and is classified into red, yellow, green and blue according to the level of their compliance. A host is red if less than 80% of its promises are kept, yellow if less than 80% of its promises were initially kept, but CFEngine was able to repair to above 80% promises kept during a run, green if more than 80% of its promises are kept, and blue if there is no contact between the hub and the client host.

Compliance Summary: The row of bar meters shows the compliance of all registered hosts¹ in blocks of 6 hours for the past week. It summarizes performance and anomalous behavior in a simple red (promises not kept), yellow (promises repaired) and green (promises kept) scale.

Services/Goals: A summary of Mission goals (as defined in user policy files; these examples are from 'company_knowledge.cf'). Edit the file in the policy editor (Planning room -> repository), or edit the file in your own text editor, to change these goals.

¹ Blue hosts will not appear here

2.1.2 Mission Engineering

Mission engineering illustrates the state of the system in relation to the desired state at all scales. Zoom in to specific areas and examine the impact of promises, query data, and extract reports.

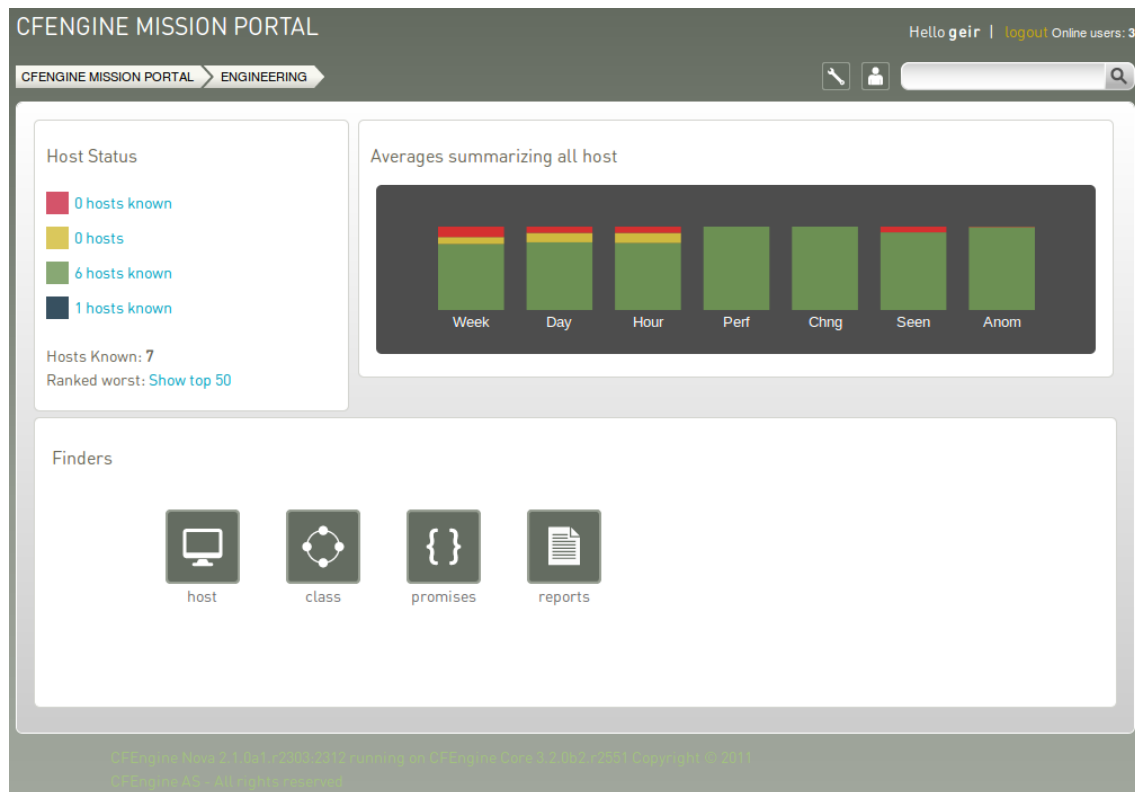


Figure: Mission Engineering

Host Status: The hosts are classified into red, yellow, green and blue according to the status of their compliance. A host is red if less than 80% of its promises are kept, yellow if less than 80% of its promises were initially kept, but CFEngine was able to repair to above 80% promises kept during a run, green if more than 80% of its promises are kept, and blue if there is no contact between the hub and the client. Clicking on the colored links produces a list of the hosts in that category.

Averages summarizing all hosts: The row of bar meters shows the compliance of all registered hosts over the past week, the past day and the past hour. It also summarizes performance and anomalous behavior in a simple red (promises not kept), yellow (promises repaired), and green (promises kept) scale.

Finders: The Mission Engineering room comes with five finder functions: host, class, promises, reports and reports summary (see section on Finders).

2.1.3 Mission Planning

Make changes to policies, goals determined by promises and implement specific tactics to achieve the desired state. Interact with data, approve changes and anomalies. Get an overview of users logged on to the Mission Portal, as well as their current activity.

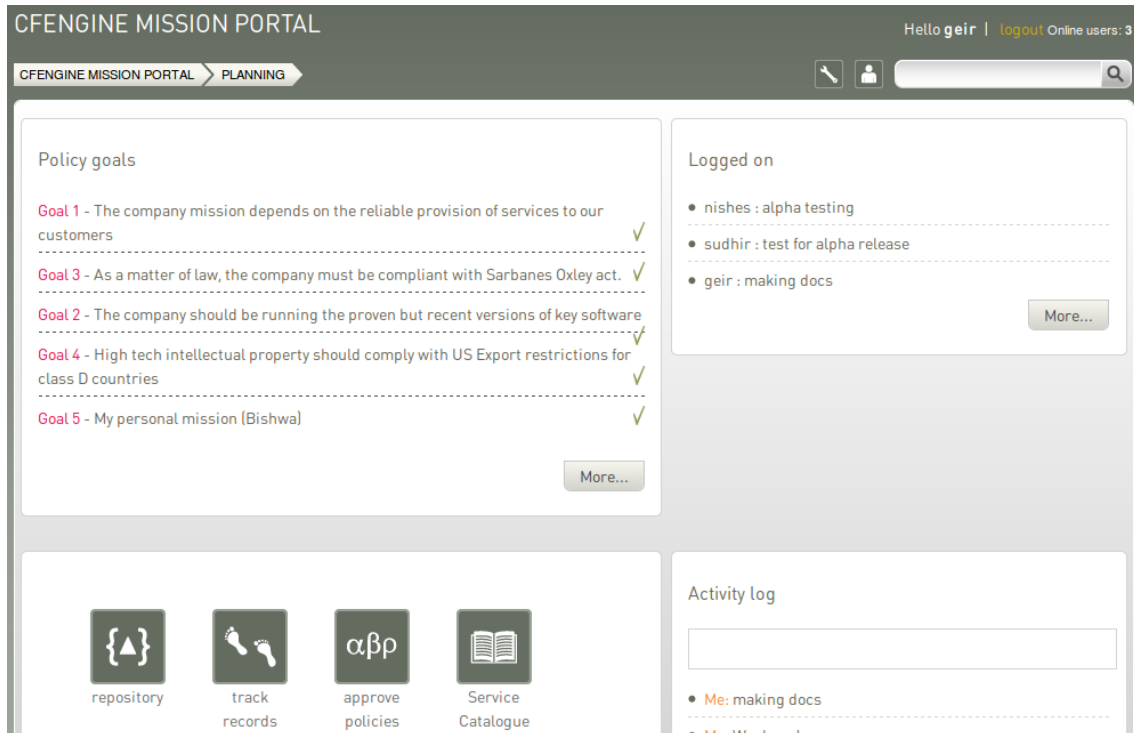


Figure: Mission Planning

Policy Goals: List of policy goals as defined in user policy files; these examples are from 'company_knowledge.cf'. Edit the file in the policy editor (Planning room -> repository) or edit the file in your own text editor. The "More..." button links to the Service Catalogue, click to see which bundles contribute to these policy goals.

Action buttons:

- Repository: Edit policy files in the integrated policy editor (requires subversion)
- Track records: To be developed
- Approve policies: To be developed
- Service catalogue: See which bundles contribute to policy goals

Logged on: Shows users currently logged on to the Mission Portal and their activity.

Activity log: Shows the latest activity entries. Type in a new activity to keep colleagues posted on current work.

2.1.4 Mission Library

The Library contains finders for documents, topics, a notes archive, and (external) link to the CFEngine community.

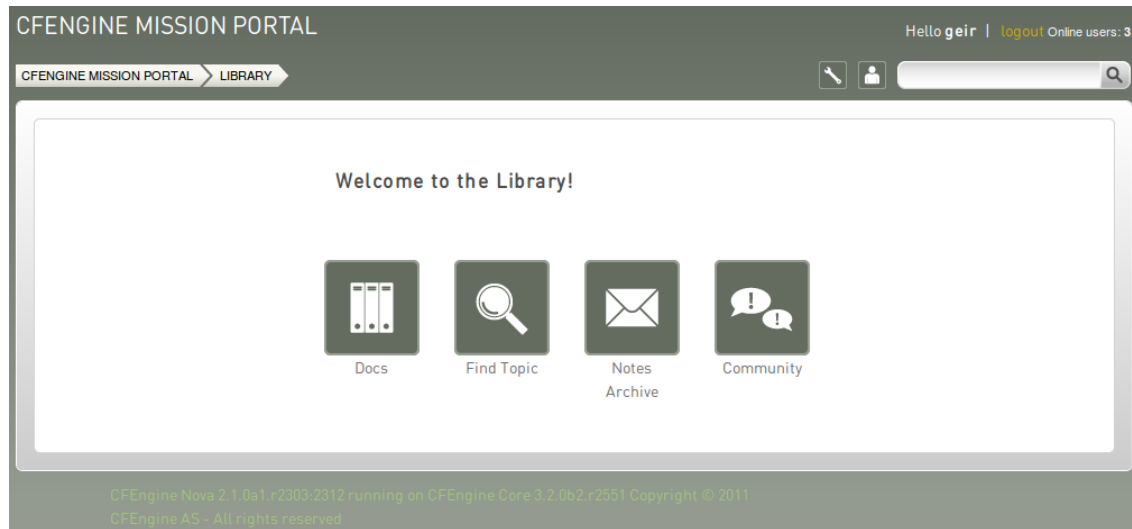


Figure: Mission Library

Library buttons:

- Docs: Overview of documentation that was packaged with CFEngine Nova.
- Find Topic: Search for topics either by scrolling through the alphabetical list or by typing in the search box (same as the search box on top right of page).
- Notes Archive: Get overview of all notes made in regard to hosts or reports.
- Community: External link to the CFEngine community

2.2 Finders

Finders are modules that make it simple and intuitive to browse and search for objects of a particular type.

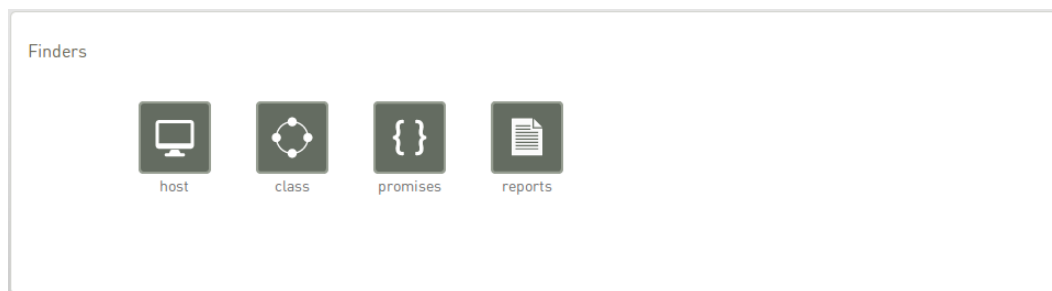


Figure: Finders in the Mission Engineering room

2.2.1 Host finder

The host finder is located in the Mission Engineering room and will display a list of hosts. Browse by scrolling through the list, click a letter corresponding to the first letter of a host name, or search for hosts in the search box (top right corner). Clicking on a host name will bring you to the host viewer.

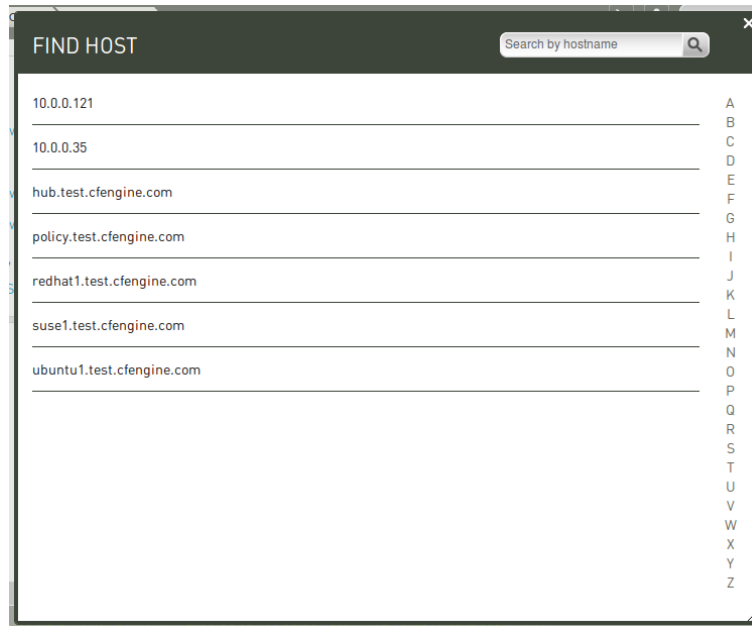


Figure: Host finder

2.2.2 Class finder

The class finder is located in the Mission Engineering room and will display a list of classes. Browse by scrolling through the list, click a letter corresponding to the first letter of a class name, or search for classes in the search box (top right corner; choose between searching all, time, soft or IP classes). Clicking on a class will bring you to a report for that class profile.

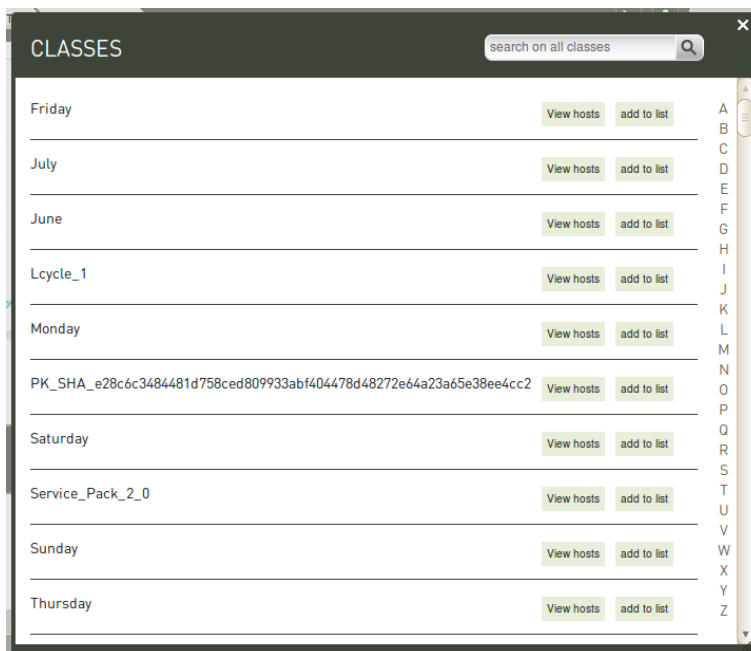


Figure: Class finder.

2.2.3 Promise finder

The promise finder is located in the Mission Engineering room and will display a list of promises. Browse by scrolling through the list, click a letter corresponding to the first letter of a promiser/bundle/handle name (set alternative in searchbox and click a letter in the right column), or search for promiser/bundle/handle in the search box (top right corner; choose between searching promiser, bundle, or handle). Clicking on a promise/bundle will bring you to the promise/bundle viewer.

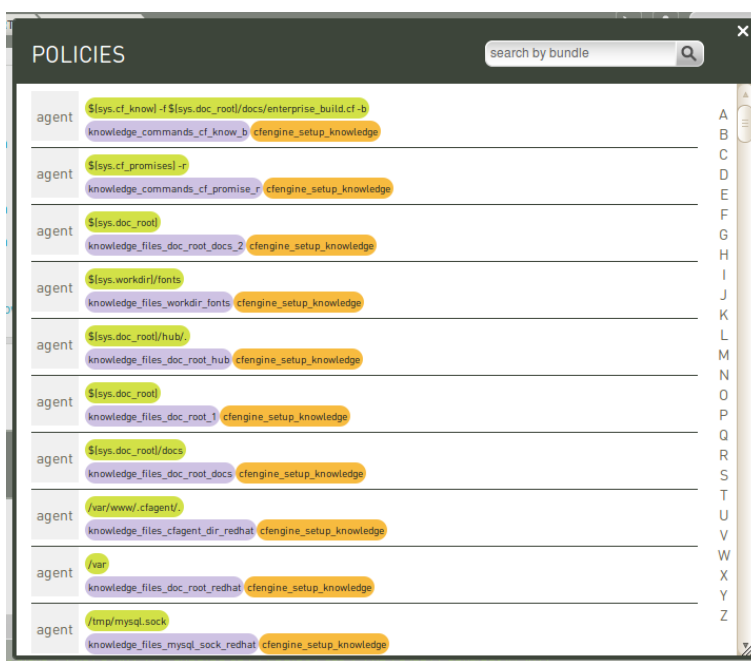


Figure: Promise finder

2.2.4 Report finder

The report finder is located in the Mission Engineering room and will display a list of standard report categories. Browse by scrolling through the list. Clicking on a report category will bring up a query table that will produce a report according to the entered query criteria.

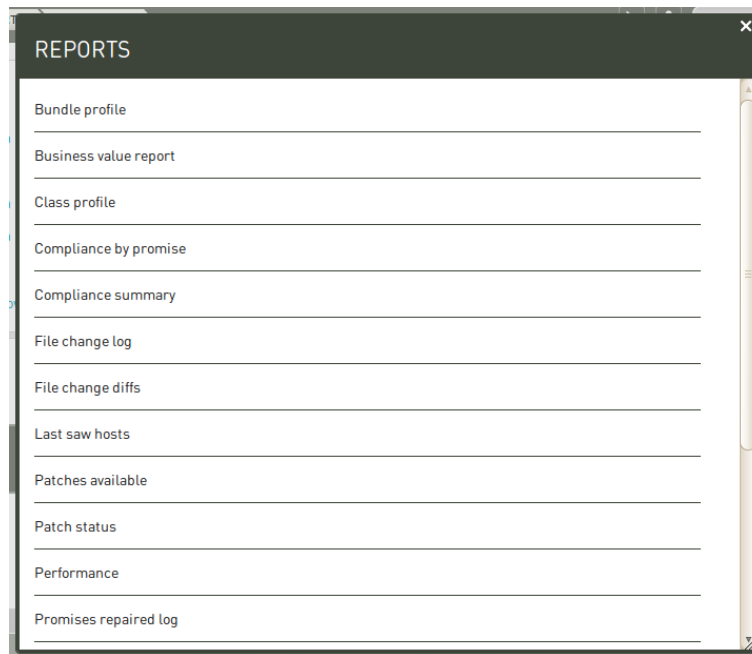


Figure: Report finder

Standard reports:

- Bundle profile: Status of promise bundles and when they were last verified
- Business value report: Accumulated value of promises kept
- Class profile: User defined classes observed on the system
- Compliance by promise: Compliance of each promise individually
- Compliance summary: Total summary of host compliance
- File change log: Log of all detected changes to files from changes promises
- File change diffs: Delta/difference comparison showing file changes
- Last saw hosts: Log of when neighbouring hosts were last observed online
- Patches available: A list of patches currently claimed to be available by the local package manager
- Patch status: A list of (un)applied patches according to the local package manager
- Performance: Execution time used to verify selected promises
- Promises repaired log: Log of actual repairs made to the system
- Promises repaired summary: Cumulative (histogram) summary of promises repaired
- Promises not kept log: Log of promises that could not or would not be kept
- Promises not kept summary: Cumulative (histogram) summary of promises not kept
- Setuid/gid root programs: Current list of observed setuid/setgid root programs

- Software installed: List of software packages claimed to be installed according to the local package manager
- Variables: Table of variable values last observed

2.2.5 Topic finder

The Topic finder is located in the Mission Library and will display a list of common topics. Browse by scrolling through the list, click a letter corresponding to the first letter of a topic name, or search for topics in the search box (top right corner). Clicking on a topic will bring you either to a document, web page or the Topic viewer (Knowledge map).

Figure: Topic finder

2.3 Viewers

Viewers show info about the main objects at different scales of the system.

2.3.1 Host viewer

Shows information about hosts, including name, status, operating system, vital signs, promises not kept, standard and custom reports, and more. View and make notes about the host.

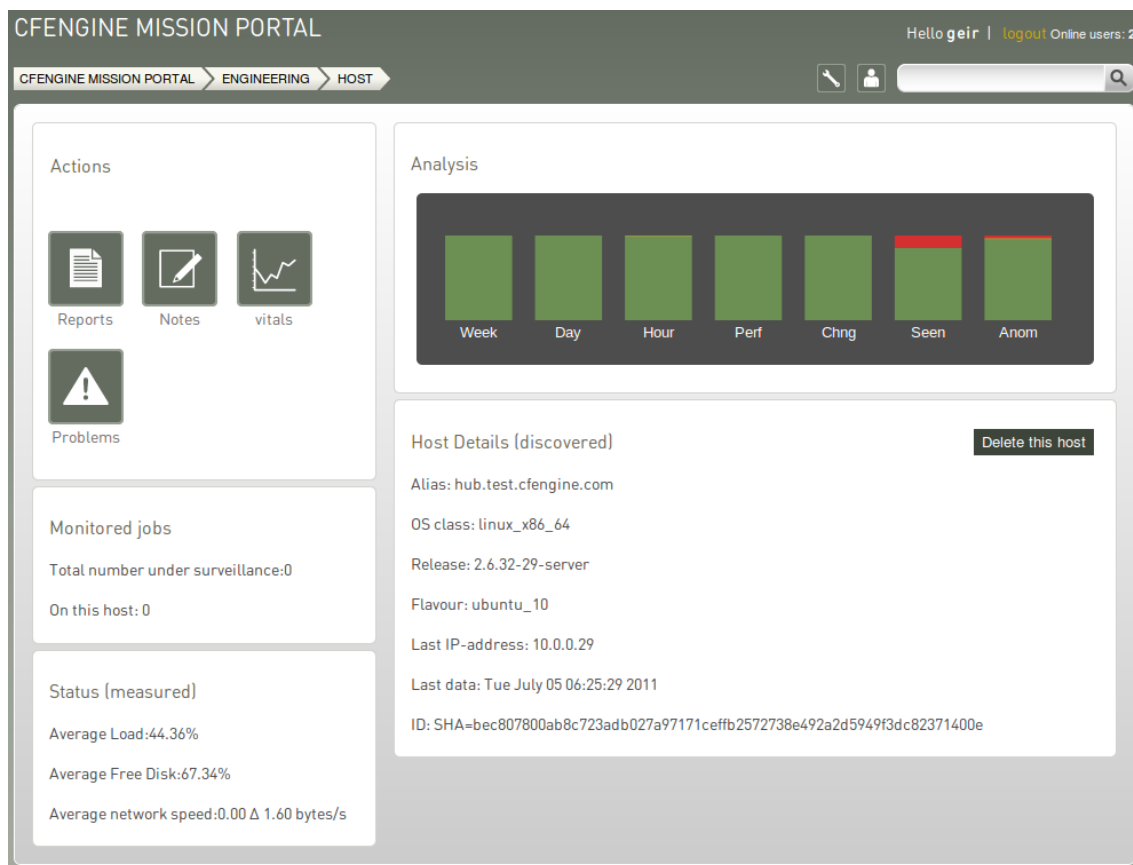


Figure: Host viewer

Action buttons:

- Reports: Tabular summary of the host's internal information, tailored to a particular topics
- Notes: View and make comments about this host
- Vitals: Overview of monitoring data for this host and its current performance statistics
- Problems: Overview of promises not kept by this host

Monitored jobs: To be developed.

Analysis: The bar meter shows the host-summary status of a number of key performance indicators.

- Week: The average level of promise-compliance over the whole past week.
- Day: The average level of promise-compliance over the past day.
- Hour: The average level of promise-compliance over the past hour.
- Perf: The average performance status of the system, compared to the learned norm.
- Chng: Software update status of the system (only shows on hub, not displayed on clients).
- Seen: The average level of connectivity compliance (to the hub) over the past week .
- Anom: Level of anomalous site-wide activity on the system.

2.3.2 Bundle viewer

The bundle viewer shows a list of classes using the bundle and the promises made within that bundle. Tabs also display other bundles using the one currently viewed or an overview of all bundles.

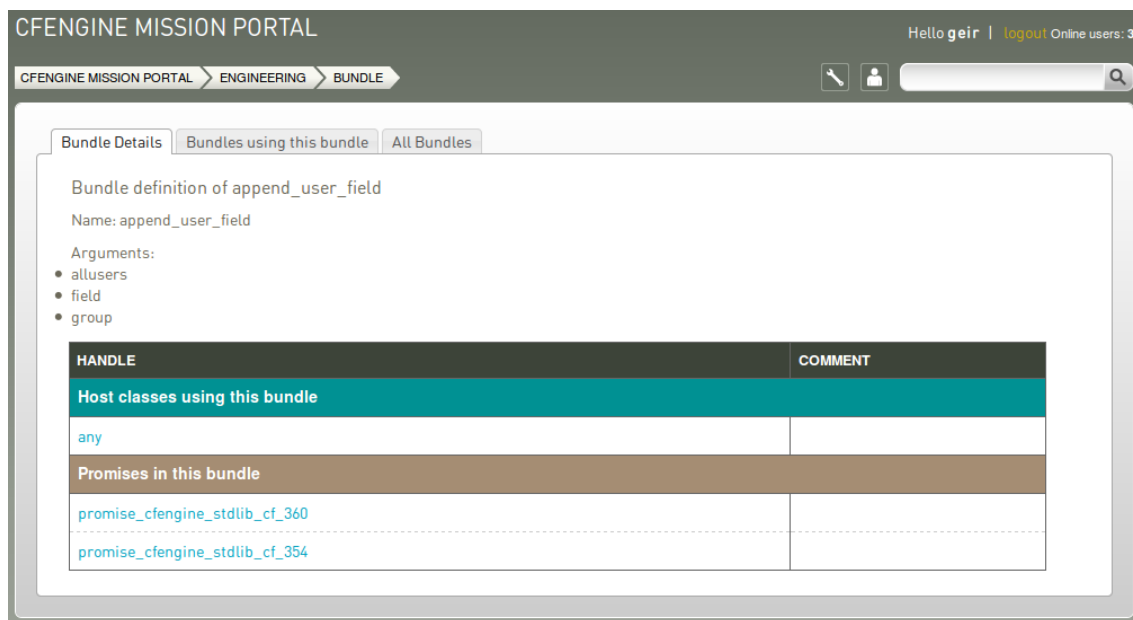


Figure: Bundle viewer

2.3.3 Promise viewer

The promise viewer shows a promise definition and body. There are tabs for viewing leads (promise type, context, dependencies), other promises used in same bundle, other promises made by same promiser, and other promises of same type.

CFENGINE MISSION PORTAL

Hello geir | [logout](#) Online users: 1

CFENGINE MISSION PORTAL > ENGINEERING > PROMISE

Definition Leads Same Bundle Same Promiser Same Type

Promise definition	
Belonging to agent bundle	cfengine_setup_knowledge
Reference handle	knowledge_files_cfagent_dir_redhat
Affected object (promiser)	/var/www/.cfagent/.
Stakeholders (promisees)	None mentioned
Comment on original intention	Ensure permissions to .cfagent directory
Promise is about	files
Applies in the class context	redhat.am_policy_hub
Defined in file	/var/cfengine/inputs/knowledge.cf
Body of the promise	
depth_search =>	recurse_basedir[inf]
perms =>	mog(700,\$[owner[redhat]],\${group[redhat]})

Figure: Promise viewer

2.3.4 Vital signs viewer

The Vital signs viewer shows an overview of monitoring data from each host and its current performance statistics. In order to see data in these graphs, each host in the CFEngine managed network must be running `cf-monitor` and `cf-serverd`. This is the default behavior for a Nova installation.



Figure: Host viewer

2.3.5 Topics viewer (Knowledge map)

The Topics viewer, or Knowledge map, is a sematic web of subject references and document pointers. It can be found by searching for a topic in the top right corner or through the topic finder in the Mission Library (this will sometimes also lead directly to a document or web page instead of the topic viewer).

Figure: Topics viewer (Knowledge map)

The yellow sphere represents the current topic, surrounding (blue) spheres represent related topics, and the size of the circles represents the number of associations each topic has. This map is navigable: click on a different topic to see a new view centered on that topic, with its related topics attached. The tabs will show leads, documents and web pages related to the current topic.

2.3.6 Weakest hosts viewer

A link to the worst ranked hosts is located in the Mission Engineering room. This ranks all hosts that have reported to the hub¹ according to their level of promises not kept, with the worst host on top. A secondary level of ranking looks at the level of promises repaired, where the host with most repairs gets a worse status.

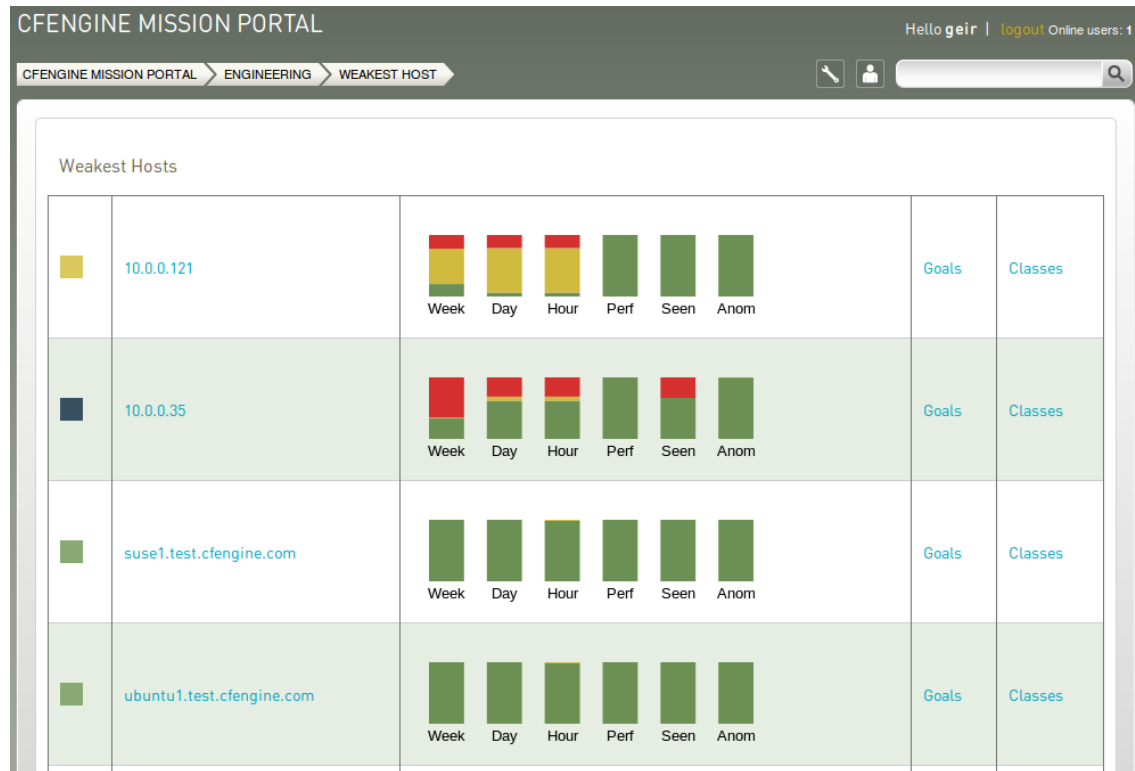


Figure: Weakest hosts

2.3.7 Report viewer

A significant capability of CFEngine Nova is the existence of automated system reporting. A report is a tabular summary of CFEngine's internal information, tailored to a particular purpose. Reports are searchable and describe attributes and qualities of managed hosts.

Figure: Report viewer

Standard reports include:

- Bundle profile: Status of promise bundles and when they were last verified
- Business value report: Accumulated value of promises kept
- Class profile: User defined classes observed on the system
- Compliance by promise: Compliance of each promise individually
- Compliance summary: Total summary of host compliance

¹ Blue hosts will not appear

- File change log: Log of all detected changes to files from changes promises
- File change diffs: Delta/difference comparison showing file changes
- Last saw hosts: Log of when neighbouring hosts were last observed online
- Patches available: A list of patches currently claimed to be available by the local package manager
- Patch status: A list of (un)applied patches according to the local package manager
- Performance: Execution time used to verify selected promises
- Promises repaired log: Log of actual repairs made to the system
- Promises repaired summary: Cumulative (histogram) summary of promises repaired
- Promises not kept log: Log of promises that could not or would not be kept
- Promises not kept summary: Cumulative (histogram) summary of promises not kept
- Setuid/gid root programs: Current list of observed setuid/setgid root programs
- Software installed: List of software packages claimed to be installed according to the local package manager
- Variables: Table of variable values last observed

2.4 Editors

2.4.1 Policy editor

The CFEngine Nova Mission Portal provides an editor for working on CFEngine language. The editor provides syntax high-lighting and look-up to make working with CFEngine's extensive language easier.

The main key commands in the editor Window are:

Autocompletion: *Ctrl+Space*

Shows a pop-up menu of possible items. This is context sensitive, e.g. it also works inside lists (e.g. `bsdflags`) to provide possible values.

Undo: *Ctrl+Z*

In Safari, Ctrl-backspace may be used.

Redo: *Ctrl+Y*

Undo an undo operation, i.e. reverse the direction of transaction roll.

Indent: *TAB*

Format a file to a standard indentation.

Multiple documents appear as tabs along the top of the screen.

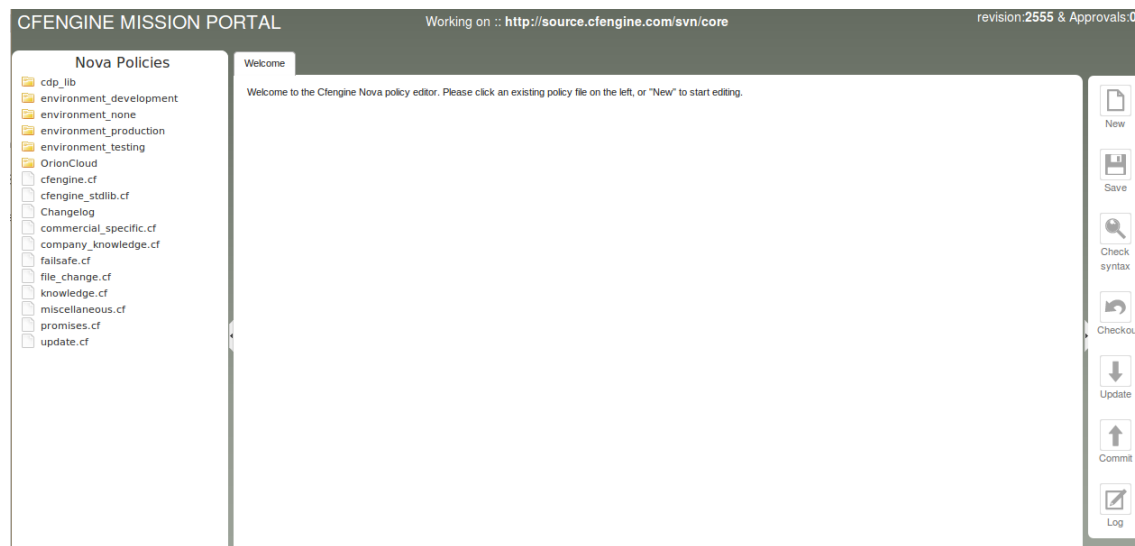


Figure: The Policy Editor

The CFEngine Nova policy editor detects syntax errors and highlights these in red to avoid mistakes when editing. In addition, by using the **Check syntax** button, it is possible to pre-test the policy before committing changes to a repository. This will run `promises.cf` through the `cf-promises` parser.

The main menu on the left hand panel shows the main workflow items for policy editing. Clicking the arrow in the panel divider collapses the menu and gives full-screen editing.

