



KBEngine

WebConsole

Guide

Table of contents

Environment Configuration.....	3
1. Requirements.....	3
2. Setup.....	3
Operation Steps.....	3
1. Start the server.....	3
2. Login system.....	4
Server Management Functions.....	5
1. User Management.....	5
2. Cluster Management.....	7
3. Performance Analysis.....	9
4. Log View.....	12

5. State Diagram.....	13
6. Python Console.....	14
7. Watcher.....	15
8. SpaceViewer.....	16

(—) Environment Configuration

1. Requirements

Python Version: Python3.X (recommended 3.3 or above)

Django Version: 1.8.9

2. Setup

- 1) First, install the matching Django module for Python. If you do not want to install Django, there are two options:
 - If you are using python2.6.6, enter into the "kbe/tools/server/django_packages" directory and extract the Django-1.6.11.tar.gz file (unzip to the current directory);
 - If you are using python2.7 or above, go to the "kbe/tools/server/django_packages" directory and extract the Django-1.8.9.tar.gz file (unzip to the current directory).
- 2) Modify sync_db.bat, sync_db.sh and run_server.bat, run_server.sh to match the path to the python you are using. On first run you need to initialize the data:

- python3.3 + django 1.8.9 under windows, run “sync_db.bat”
- python3.3 + django 1.8.9 under linux, run “sync_db.sh”;
- python2.6 + django 1.6.11 under linux, run “sync_db_dj-1.6.sh”;
- python2.6 + django 1.6.1 under windows, please refer to “sync_db_dj-1.6.sh” and build a .bat file.

(二) Operation Steps

1. Start the server

- Linux: Run the run_server.sh script, or deploy to nginx. In a browser, open “<http://xxx.xxx.xxx.xxx:8000/wc/>” for access where “xxx.xxx.xxx.xxx” is the Linux machine’s IP address;
- Windows: Run run_server.bat (make sure to change python folder in run_server.bat to your own python folder) and open “<http://xxx.xxx.xxx.xxx:8000/wc/>” in a web browser, where “xxx.xxx.xxx.xxx” is the machine’s IP address.

2. Login System

The initial login requires a default username and password to login to the user management interface and create a new administrative user:

- When using the Web Console for the first time, the default login account is “Admin” and the default password is “123456”, this account is also the only background administrative account. Please promptly change your password after login.
- The first time you use the Admin account to enter the background, you need to create a server management account using your user account name and UID. After creating a new user account, log out of Admin and log into the new user.

- The background environment is python3.3 + django 1.8.9, and python2.6.6 + django-1.6.11 under linux. Tests passed.
- All functions of the Web Console are derived from the KBEEngine server. Therefore, to use the functions of the console, it must be ensured that the server process runs correctly.
- If you have any questions, please ask them on the KBEEngine official forum.

KBEEngine web console

账 号

密 码

登录

2-1-1 登录界面

(三) Server Management Functions

1. User Management

1) Account Management

On this page, you can manage users who use the Web Console, or manage Administrator accounts.

KBEEngine控制台								用户名: Admin	退出
账号管理	Id	账号名	显示名	操作系统用户	操作系统用户id	kbe_root	kbe_res_path	kbe_bin_path	操作
新建账号	0	Admin	Admin	UNKNOWN	-1				修改资料 修改密码 删除

3-1-1-1 Account Management Interface

2) Management user creation

- Account name: Login account;
- Nickname: Displayed after login;
- Login password: Any combination of alphanumeric characters;
- Confirm password: Enter the password again;
- Operating System user: Linux system user name who is running KBE server. Please ignore under Windows.
- Operating System uid: Linux system user uid. Make sure to enter the uid of the user running KBE server, otherwise it cannot be managed. Please ignore under Windows.
- KBE_ROOT: The KBE_ROOT directory. Defaults to the root of the current Web Console. (can be empty)
- KBE_RES_PATH: The KBE_RES_PATH directory. Defaults to the root of the current Web Console. (can be empty)
- KBE_BIN_PATH: The KBE_BIN_PATH directory. Defaults to the root of the current Web Console. (can be empty)

Note: Because there can be multiple KBEs on a single server, each managed system uses User, UID, KBE_ROOT, KBE_RES_PATH, KBE_BIN_PATH and cannot be referenced by other users.

The screenshot shows the 'KBEEngine控制台' (KBEEngine Control Panel) interface. On the left, there is a sidebar with '账号管理' (Account Management) and '新建账号' (New Account). The main area displays the '添加新用户' (Add New User) form. The form includes the following fields:

- 账号 (Account Number):
- 昵称 (Nickname):
- 登录密码 (Login Password):
- 确认密码 (Confirm Password):
- 操作系统用户 (OS User):
- 操作系统用户uid (OS User UID):
- KBE_ROOT:
- KBE_RES_PATH:
- KBE_BIN_PATH:

At the bottom right of the form is a button labeled '添加' (Add).

3-1-2-1 New account

3) Modify user

Enter the new user properties and click OK to modify.

The screenshot shows the 'KBEEngine控制台' (KBEEngine Control Panel) interface with a table of users. A modal dialog box titled '账号编辑' (Account Edit) is open, allowing for user modification. The dialog includes the following fields:

- 修改用户 (Modify User):
- 新昵称 (New Nickname):
- 操作系统用户 (OS User):
- 操作系统用户uid (OS User UID):
- KBE_ROOT:
- KBE_RES_PATH:
- KBE_BIN_PATH:

At the bottom of the dialog is a button labeled '确认修改' (Confirm Modify).

3-1-3-1 Account editing

4) Change password

Enter the new password twice and click Modify.

KBEngine控制台

账号管理

新建账号

密码修改

账号名

whw

密码第一次

密码第二次

修改

3-1-4-1 Password Modification

2. Cluster Management

1) Server Management

In the “Server Management” page, you can manage and view resource consumption of started KBE processes.

- STOP operation: Stop current process.
- KILL operation: Kill current process.
- Start new component: Start new KBE service or component.
- Stop the server: Stop the current KBE server and all processes.
- Save current server configuration: Save the current KBE server process configuration.

KBEngine控制台

用户名: RM 退出

集群管理

服务器管理

所有守护进程状态

服务器运行配置

性能分析

日志查看

状态图

Python控制台

Watcher

SpaceViewer

刷新

启动新组件

停止服务器

保存当前服务器运行配置

Machine	组件名称	uid	pid	cid	gid	gus	CPU负载	内存消耗比	内存消耗数	实体数量	Proxy实体数量	客户端数量	操作
192.168.191.1	baseappmgr	519	8572	62915000	1	5	0.00%	0.09%	15m	0	0	0	STOP KILL
192.168.191.1	cellappmgr	519	6172	62916000	2	6	0.00%	0.09%	15m	0	0	0	STOP KILL
192.168.191.1	dbmgr	519	8560	62914000	4	4	0.00%	0.17%	28m	0	0	0	STOP KILL
192.168.191.1	cellapp1	519	9164	62918001	5	9	0.00%	0.69%	112m	1939	0	0	STOP KILL
192.168.191.1	baseapp1	519	7464	62917001	3	7	0.00%	0.25%	40m	121	2	1	STOP KILL
192.168.191.1	loginapp	519	7180	62919000	6	11	0.00%	0.14%	22m	0	0	0	STOP KILL
192.168.191.1	logger	519	8628	62912000	1	2	0.00%	0.14%	23m	0	0	0	STOP KILL
192.168.191.1	interfaces	519	8380	62913000	1	3	0.00%	0.17%	27m	0	0	0	STOP KILL

3-2-1-1 Server Management Interface

2) Start new component

On this page you can create any number of component processes within the server cluster.

KBEngine控制台

集群管理

服务器管理

所有守护进程状态

服务器运行配置

性能分析

日志查看

状态图

Python控制台

Watcher

SpaceViewer

KBEngine组件将使用以下用户启动

用户名	csfs
用户ID	519
kbe_root	
kbe_res_path	
kbe_bin_path	

启动

组件名

cellapp

运行机器

192.168.191.1

启动数量

1

启动

3-2-2-1 Start new component interface

3) All daemon status

Here you can view machine information and resources for all KBE processes in the server cluster.

KBEngine控制台

用户名: RM 退出

集群管理

服务器管理

所有守护进程状态

服务器运行配置

性能分析

日志查看

状态图

Python控制台

Watcher

SpaceViewer

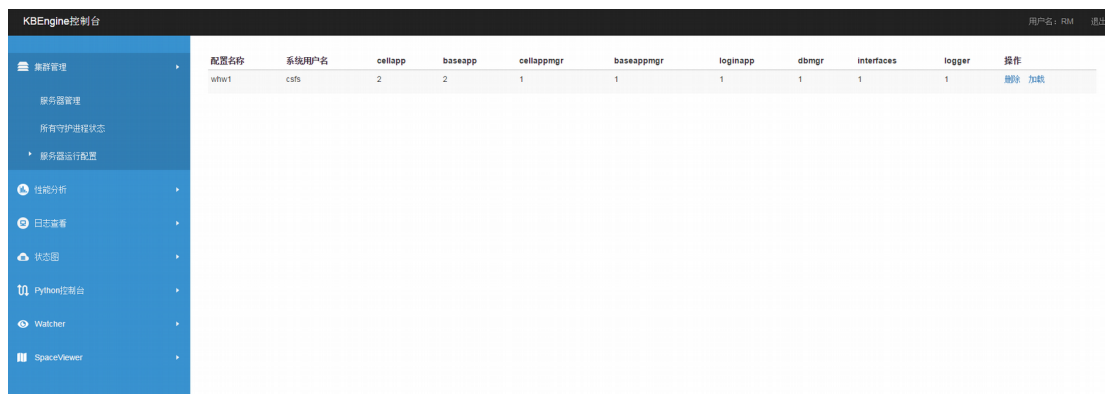
Machine	%CPU	%MEM	%pCPU	%pMem	totalMem
192.168.191.1	50.97	30.22%	1	16m	4921m/16284m

Machine	组件名称	uid	pid	cid	gid	gus	CPU负载	内存消耗比	内存消耗数	实体数量	Proxy实体数量	客户端数量
192.168.191.1	baseappmgr	519	8572	62915000	1	5	0.00%	0.09%	15m	0	0	0
192.168.191.1	cellappmgr	519	6172	62916000	2	6	0.32%	0.09%	15m	0	0	0
192.168.191.1	dbmgr	519	8560	62914000	4	4	0.00%	0.17%	28m	0	0	0
192.168.191.1	cellapp1	519	9164	62918001	5	9	4.51%	0.69%	112m	1939	0	0
192.168.191.1	baseapp1	519	7464	62917001	3	7	0.00%	0.25%	40m	121	2	1
192.168.191.1	loginapp	519	7180	62919000	6	11	0.00%	0.14%	22m	0	0	0
192.168.191.1	logger	519	8628	62912000	1	2	0.00%	0.14%	23m	0	0	0
192.168.191.1	interfaces	519	8380	62913000	1	3	0.00%	0.17%	27m	0	0	0

3-2-3-1 All daemon status

4) Server configurations

On this page you can load and delete previously saved server configurations.

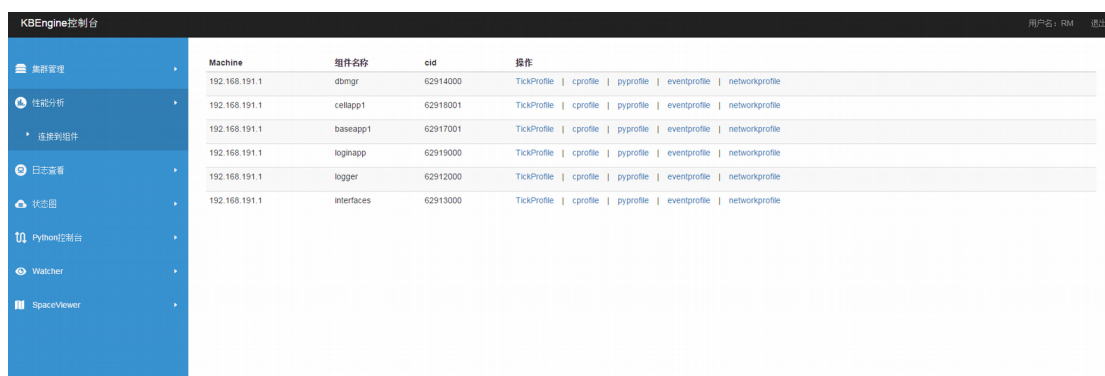


配置名称	系统用户名	cellapp	baseapp	cellappmgr	baseappmgr	loginapp	dbmgr	interfaces	logger	操作
whw1	csts	2	2	1	1	1	1	1	1	删除 加载

3-2-4-1 Server configurations

3. Performance Analysis

Here you can select the component process you want to analyze.

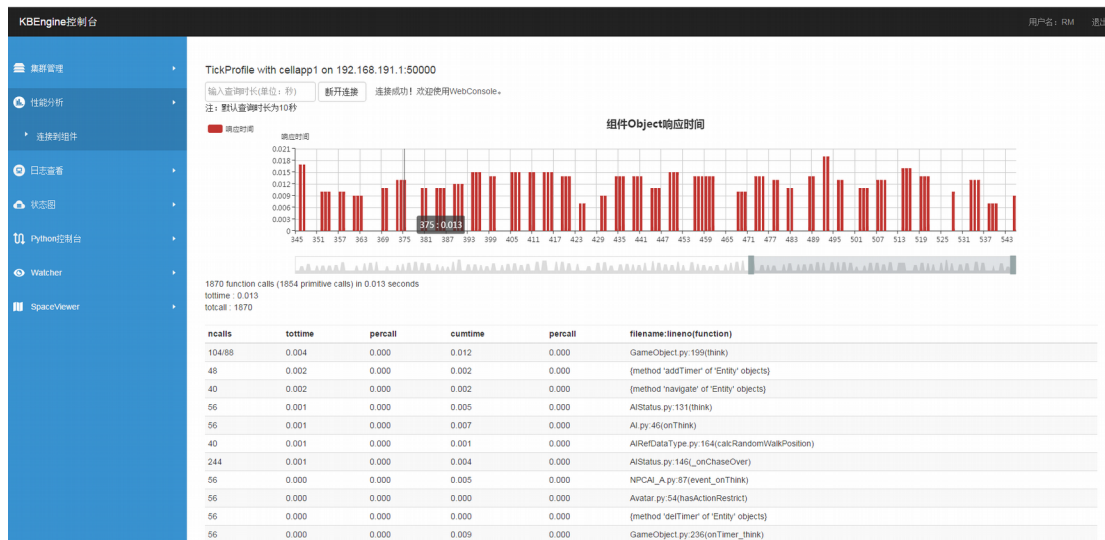


Machine	组件名称	cid	操作
192.168.191.1	dbmgr	62914000	TickProfile cprofile pyprofile eventprofile networkprofile
192.168.191.1	cellapp1	62918001	TickProfile cprofile pyprofile eventprofile networkprofile
192.168.191.1	baseapp1	62917001	TickProfile cprofile pyprofile eventprofile networkprofile
192.168.191.1	loginapp	62919000	TickProfile cprofile pyprofile eventprofile networkprofile
192.168.191.1	logger	62912000	TickProfile cprofile pyprofile eventprofile networkprofile
192.168.191.1	interfaces	62913000	TickProfile cprofile pyprofile eventprofile networkprofile

3-3-0-1 Component process selection screen

1) TickProfile

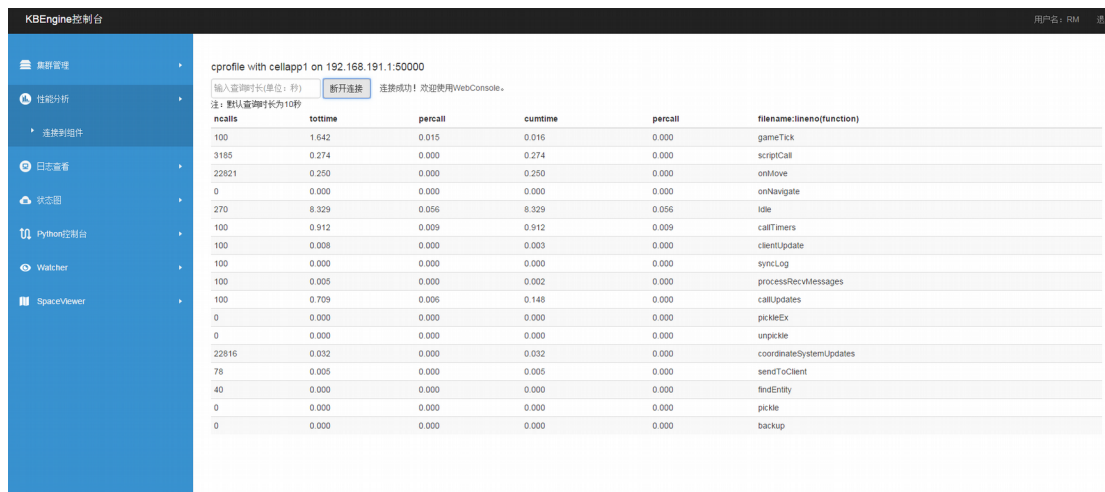
On this page you can query the tick profile. The query duration is empty, with a default length of 10 seconds. Click the bars on the graph to see details.



3-3-1-1 TickProfile screen

2) Cprofile

Here you can perform cprofile analysis queries. The query duration is empty with a default length of 10 seconds.



3-3-2-1 Cprofile screen

3) Pyprofile

Here you can perform pyprofile analysis queries. The query duration is empty with a default length of 10 seconds.

pyprofile with cellapp1 on 192.168.191.1:50000

注: 默认查询时长为10秒

ncalls	tottime	pcall	cumtime	pcall	filename:lineno(function)
10500/8380	0.303	0.000	0.938	0.000	GameObject.py:199(think)
5247	0.173	0.000	0.173	0.000	(method 'addTimer' of 'Entity' objects)
3133	0.117	0.000	0.117	0.000	(method 'navigate' of 'Entity' objects)
5253	0.098	0.000	0.404	0.000	AIStatus.py:131(think)
5253	0.075	0.000	0.562	0.000	AI.py:46(onThink)
3133	0.041	0.000	0.050	0.000	AIRefDataType.py:164(calcRandomWalkPosition)
25474	0.037	0.000	0.246	0.000	AIStatus.py:146(_onChaseOver)
5253	0.035	0.000	0.438	0.000	NPCAI_A.py:87(event_onThink)
5253	0.027	0.000	0.027	0.000	(method 'delTimer' of 'Entity' objects)
5253	0.026	0.000	0.026	0.000	Avatar.py:54(hasActionRestrict)
5253	0.025	0.000	0.759	0.000	GameObject.py:236(onTimer_think)
5253	0.023	0.000	0.023	0.000	Avatar.py:818(isDead)
22347	0.018	0.000	0.026	0.000	ECBExtend.py:111(onMove)
8380	0.011	0.000	0.013	0.000	random.py:342(uniform)
5403	0.009	0.000	0.775	0.000	ECBExtend.py:96(onTimer)
5253	0.006	0.000	0.568	0.000	NPC.py:77(onThink)
16766	0.003	0.000	0.003	0.000	(method 'random' of '_random.Random' objects)
3127	0.003	0.000	0.244	0.000	ECBExtend.py:131(onMoveOver)
25474	0.002	0.000	0.002	0.000	(built-in method callable)
5247	0.002	0.000	0.002	0.000	(time)
3133	0.002	0.000	0.002	0.000	(built-in method cos)

3-3-3-1 PyProfile screen

4) Eventprofile

Here you can analyze event profile information. The query duration is empty with a default length of 10 seconds.

eventprofile with cellapp1 on 192.168.191.1:50000

注: 默认查询时长为10秒

name	count	size
Event Type PrivateClientEvents		
Player_pingBack	19	3
Player_triggerFightResultFS	17	3
Event Type PublicClientEvents		
Player_removeBuffFS	13	1
Player_seeSpellEffectFS	15	3
NPC_effectStatus	9	1
NPC_actionRestrict	9	1
NPC_MP	4	7
NPC_direction	4	1
Player_updateBuffFS	15	8
Player_startSpellFS	4	7

3-3-4-1 EventProfile screen

5) Networkprofile

Here you can analyze network performance information. The query duration is empty with a default length of 10 seconds.

name	sent#	size	avg	total#	totalsize	recv#	size	avg	total#	totalsize
Cellapp:lookApp	0	0	0	0	0	8	16	2	1341	2
Cellappmgr: updateCellapp	100	2200	22	75965	1671670	0	0	0	0	0
Logger:writeLog	9	848	96	5151	498777	0	0	0	0	0
Cellapp: onAppActiveTick	0	0	0	0	0	3	42	14	1268	1

3-3-4-1 NetWorkProfile screen

4. Log View

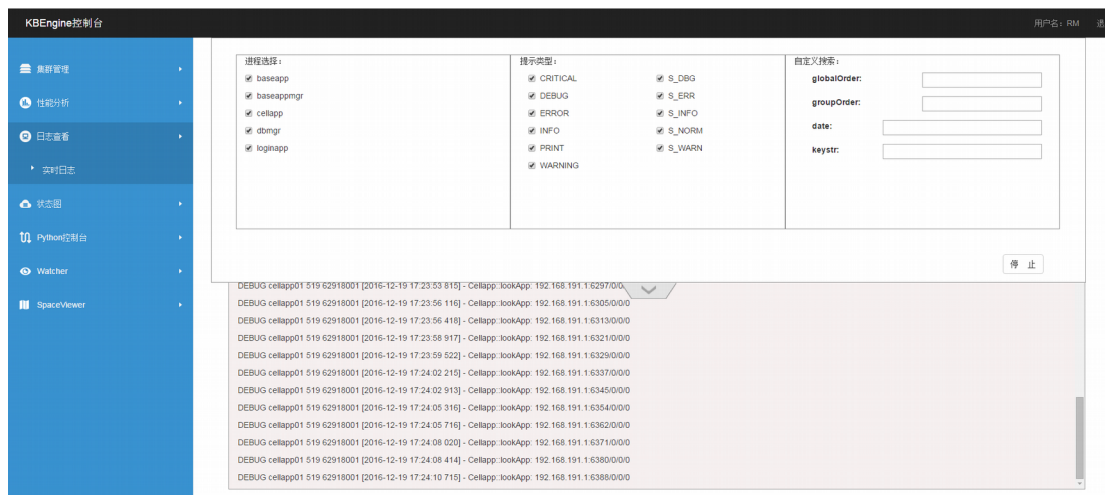
1) Real-time log

This page provides real-time log viewing and filtering. Click on the arrow for a drop down filtering menu. Use this page to view log data from all KBEEngine processes.

```

DEBUG cellapp01 519 62918001 [2016-12-19 17:22:48 413] - Cellapp:lookApp: 192.168.191.1:58010/0/0
DEBUG cellapp01 519 62918001 [2016-12-19 17:22:48 020] - Cellapp:lookApp: 192.168.191.1:57930/0/0
DEBUG cellapp01 519 62918001 [2016-12-19 17:22:40 414] - Cellapp:lookApp: 192.168.191.1:57500/0/0
DEBUG cellapp01 519 62918001 [2016-12-19 17:22:42 813] - Cellapp:lookApp: 192.168.191.1:57580/0/0
DEBUG cellapp01 519 62918001 [2016-12-19 17:22:37 718] - Cellapp:lookApp: 192.168.191.1:57290/0/0
DEBUG cellapp01 519 62918001 [2016-12-19 17:22:43 114] - Cellapp:lookApp: 192.168.191.1:57660/0/0
DEBUG cellapp01 519 62918001 [2016-12-19 17:22:50 714] - Cellapp:lookApp: 192.168.191.1:58090/0/0
DEBUG cellapp01 519 62918001 [2016-12-19 17:22:45 719] - Cellapp:lookApp: 192.168.191.1:57840/0/0
DEBUG cellapp01 519 62918001 [2016-12-19 17:22:40 116] - Cellapp:lookApp: 192.168.191.1:57420/0/0
DEBUG cellapp01 519 62918001 [2016-12-19 17:22:58 622] - Cellapp:lookApp: 192.168.191.1:58670/0/0
DEBUG cellapp01 519 62918001 [2016-12-19 17:22:53 614] - Cellapp:lookApp: 192.168.191.1:58330/0/0
DEBUG cellapp01 519 62918001 [2016-12-19 17:22:56 214] - Cellapp:lookApp: 192.168.191.1:58490/0/0
DEBUG cellapp01 519 62918001 [2016-12-19 17:23:01 214] - Cellapp:lookApp: 192.168.191.1:58800/0/0
DEBUG cellapp01 519 62918001 [2016-12-19 17:22:51 012] - Cellapp:lookApp: 192.168.191.1:58170/0/0
DEBUG cellapp01 519 62918001 [2016-12-19 17:22:55 915] - Cellapp:lookApp: 192.168.191.1:58410/0/0
DEBUG cellapp01 519 62918001 [2016-12-19 17:22:58 517] - Cellapp:lookApp: 192.168.191.1:58580/0/0
DEBUG cellapp01 519 62918001 [2016-12-19 17:23:01 517] - Cellapp:lookApp: 192.168.191.1:58880/0/0
DEBUG cellapp01 519 62918001 [2016-12-19 17:22:53 314] - Cellapp:lookApp: 192.168.191.1:58250/0/0
INFO cellapp01 519 62918001 [2016-12-19 17:23:04 767] - TeletServer: onTeletHandlerClosed: del handler(192.168.191.1:5468)
DEBUG cellapp01 519 62918001 [2016-12-19 17:23:04 120] - Cellapp:lookApp: 192.168.191.1:59050/0/0
DEBUG cellapp01 519 62918001 [2016-12-19 17:23:03 815] - Cellapp:lookApp: 192.168.191.1:58970/0/0
DEBUG cellapp01 519 62918001 [2016-12-19 17:23:06 515] - Cellapp:lookApp: 192.168.191.1:59400/0/0
DEBUG cellapp01 519 62918001 [2016-12-19 17:23:06 817] - Cellapp:lookApp: 192.168.191.1:59520/0/0
  
```

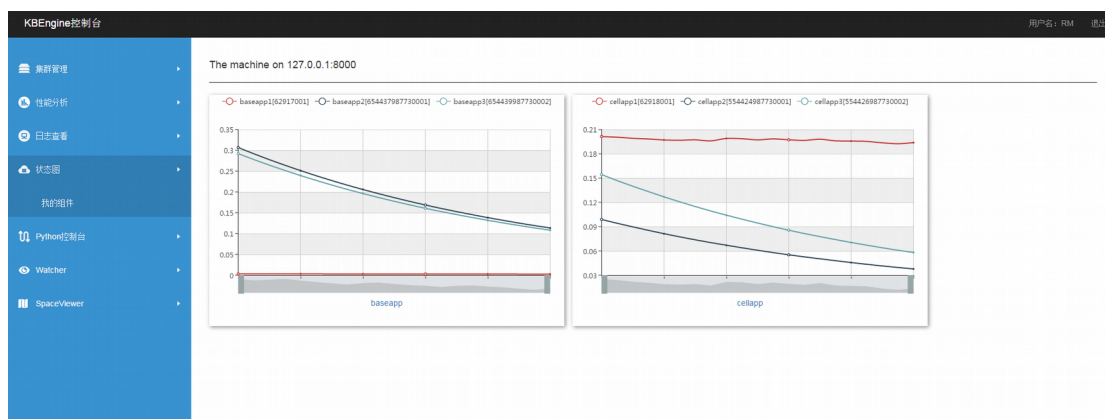
3-4-1-1 Real-time log interface



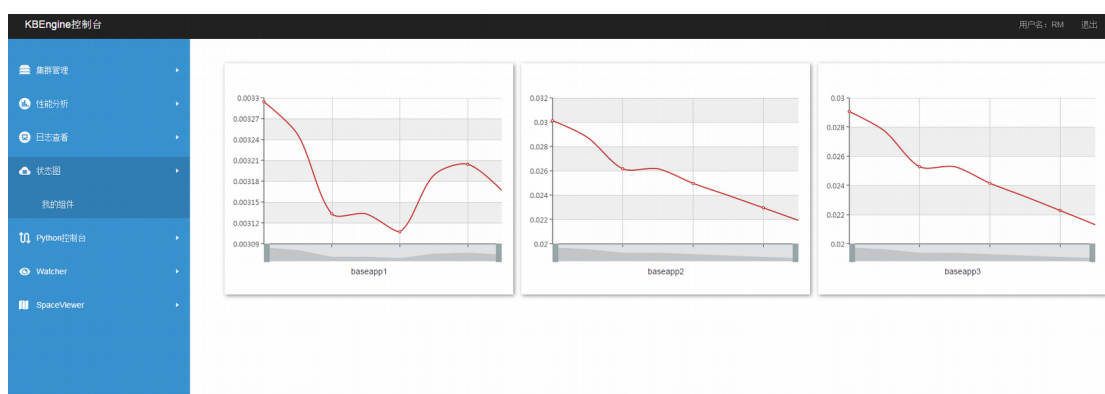
3-4-1-2 Real-time log filtering

5. State Diagram

The State Diagram page provides a linear chart of the current state of cellapp and baseapp. Click the name to see line graphs for each components individual processes.



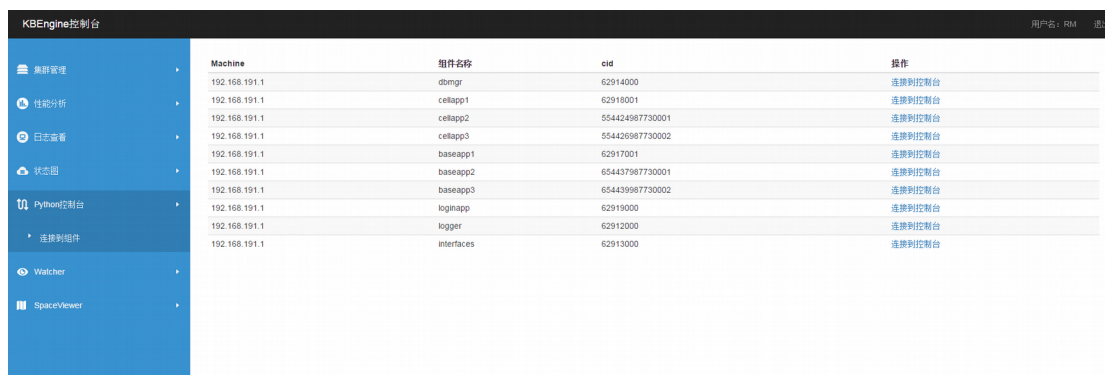
3-5-1-1 Component overview page



3-5-1-2 Component processes page

6. Python Console

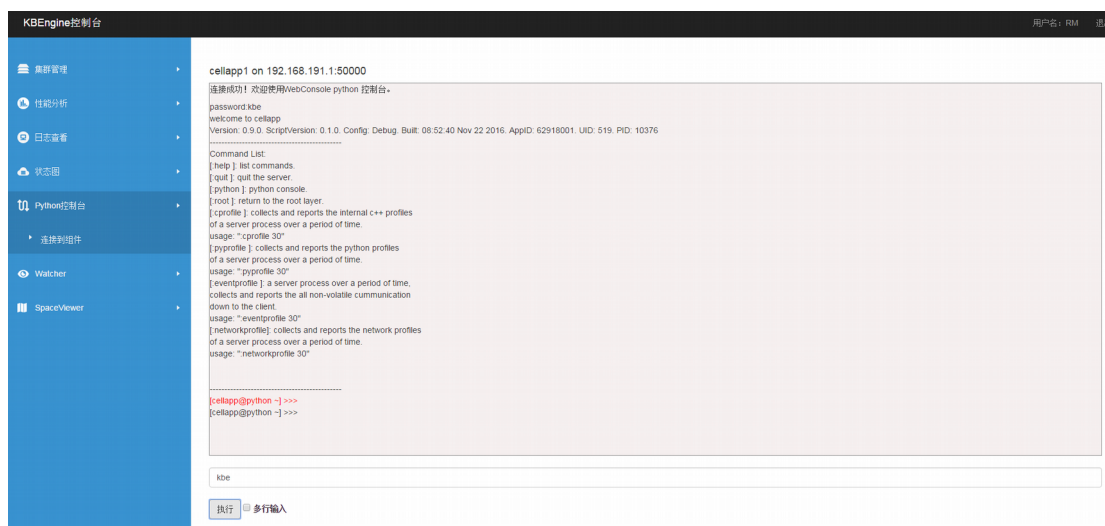
On this page you can connect a python console (through telnet) to a single process and enter commands to be executed by clicking the send button on the console page.



The screenshot shows the KBEEngine control console interface. On the left is a blue sidebar with navigation options: 集群管理, 性能分析, 日志查看, 状态图, Python控制台 (selected), 连接到组件, Watcher, and SpaceViewer. The main area displays a table of components with columns: Machine, 组件名称, cid, and 操作. The table lists various components like dbmgr, cellapp1, cellapp2, cellapp3, baseapp1, baseapp2, baseapp3, loginapp, logger, and interfaces, each with a corresponding '连接到控制台' (Connect to console) link.

Machine	组件名称	cid	操作
192.168.191.1	dbmgr	62914000	连接到控制台
192.168.191.1	cellapp1	62918001	连接到控制台
192.168.191.1	cellapp2	554424987730001	连接到控制台
192.168.191.1	cellapp3	554426987730002	连接到控制台
192.168.191.1	baseapp1	62917001	连接到控制台
192.168.191.1	baseapp2	654437987730001	连接到控制台
192.168.191.1	baseapp3	654439987730002	连接到控制台
192.168.191.1	loginapp	62919000	连接到控制台
192.168.191.1	logger	62912000	连接到控制台
192.168.191.1	interfaces	62913000	连接到控制台

3-6-1-1 Python Console Process Selection Page



The screenshot shows the KBEEngine control console interface with the Python Console component selected. The main area displays the connection status for 'cellapp1 on 192.168.191.1:50000'. It shows a successful connection message and a list of commands that can be executed in the python console, including help, quit, python, root, cprofile, pyprofile, eventprofile, networkprofile, and networkprofile. Below the command list is a text input field with the value 'kbe' and a '执行' (Execute) button.

```
cellapp1 on 192.168.191.1:50000
连接成功! 欢迎使用WebConsole python 控制台。
password:kbe
welcome to cellapp
Version: 0.9.0. ScriptVersion: 0.1.0. Config: Debug. Build: 08:52:40 Nov 22 2016. AppID: 62918001. UID: 519. PID: 10376

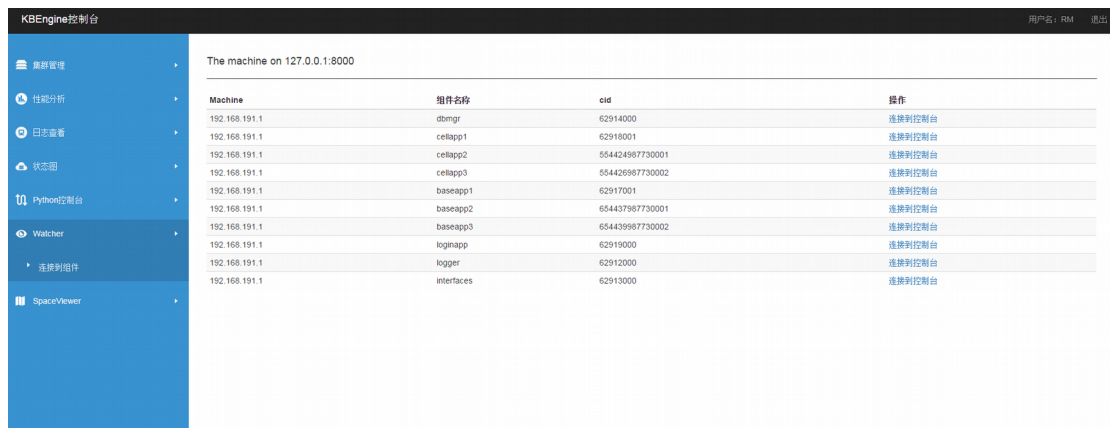
Command List:
[ help ] list commands.
[ quit ] quit the server.
[ python ] python console.
[ root ] return to the root layer.
[ cprofile ] collects and reports the internal c++ profiles
of a server process over a period of time.
usage: "cprofile 30"
[ pyprofile ] collects and reports the python profiles
of a server process over a period of time.
usage: "pyprofile 30"
[ eventprofile ] a server process over a period of time,
collects and reports the all non-volatile communication
down to the client.
usage: "eventprofile 30"
[ networkprofile ] collects and reports the network profiles
of a server process over a period of time.
usage: "networkprofile 30"

[cellapp@python ~] >>>
[cellapp@python ~] >>>
```

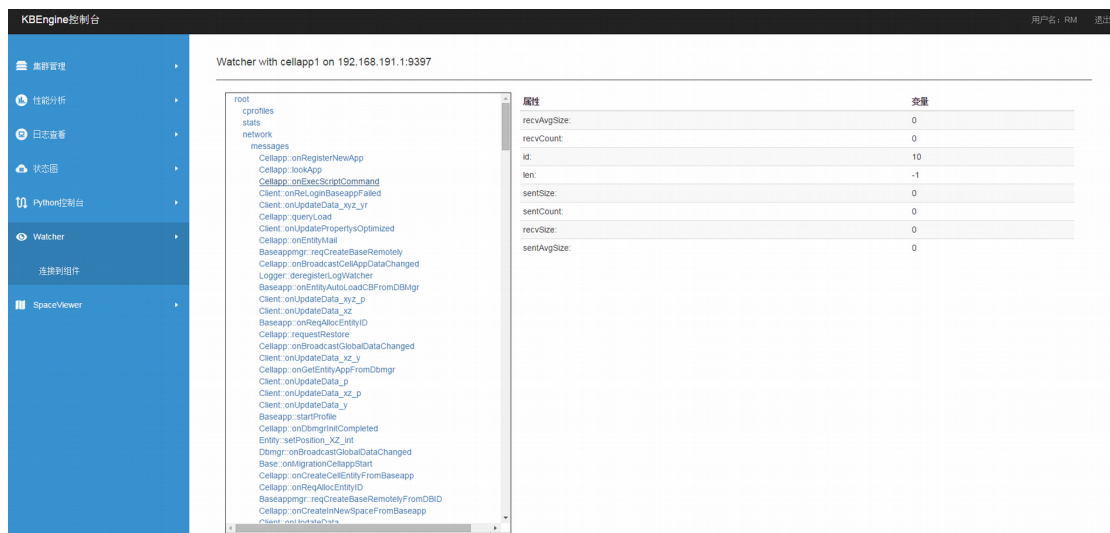
3-6-1-2 Python Console Page

7. Watcher

The watcher provides developers with a single view of all the status information for a single component process (attributes, response speed, etc.) and updates the data to the process watcher page in real time.



3-7-1-1 Watcher Process Selection Page



3-7-1-2 Watcher Operation Page

8. SpaceViewer

In SpaceViewer, you can see the distribution of entities in all Spaces in the game. The space list on the left is updated as new spaces are created. The selected Space is displayed in real time on the right, showing its state and distribution.

Note: Since the size of the space map cannot be obtained independently, the value of the XY axis of the SpaceViewer is determined by the maximum X and Y values of all entities.

