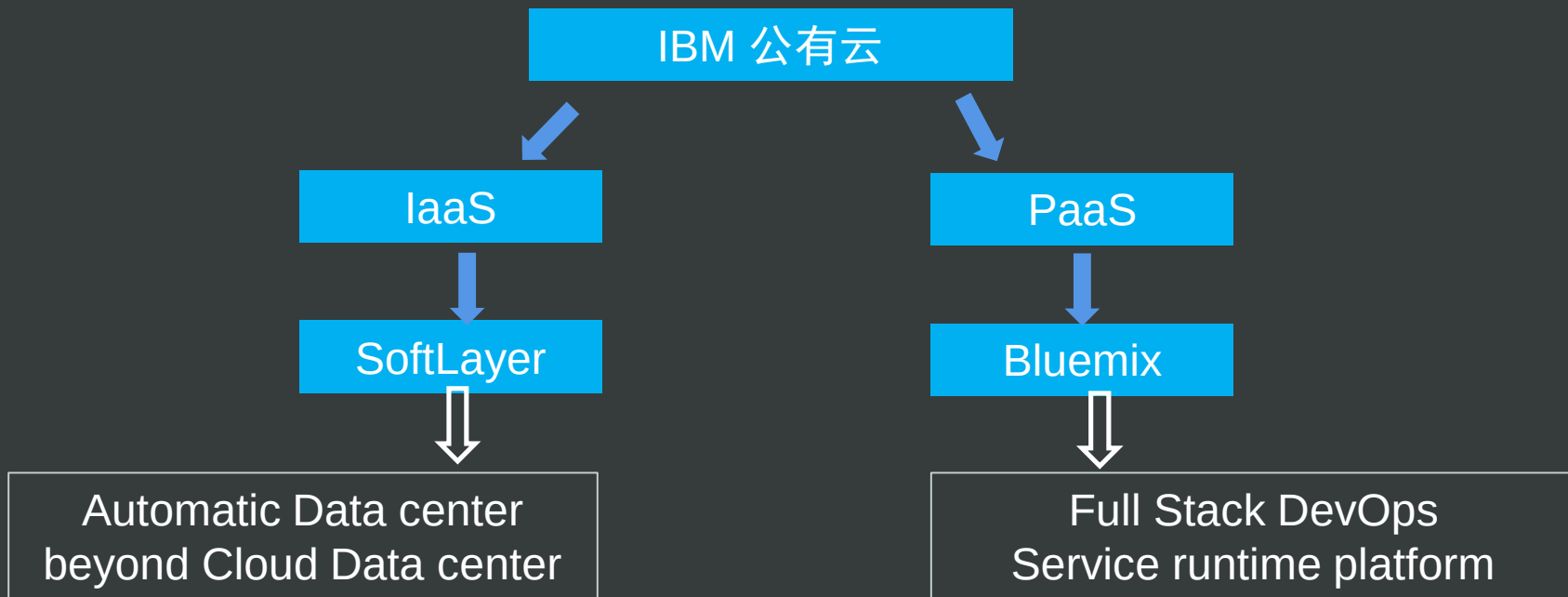


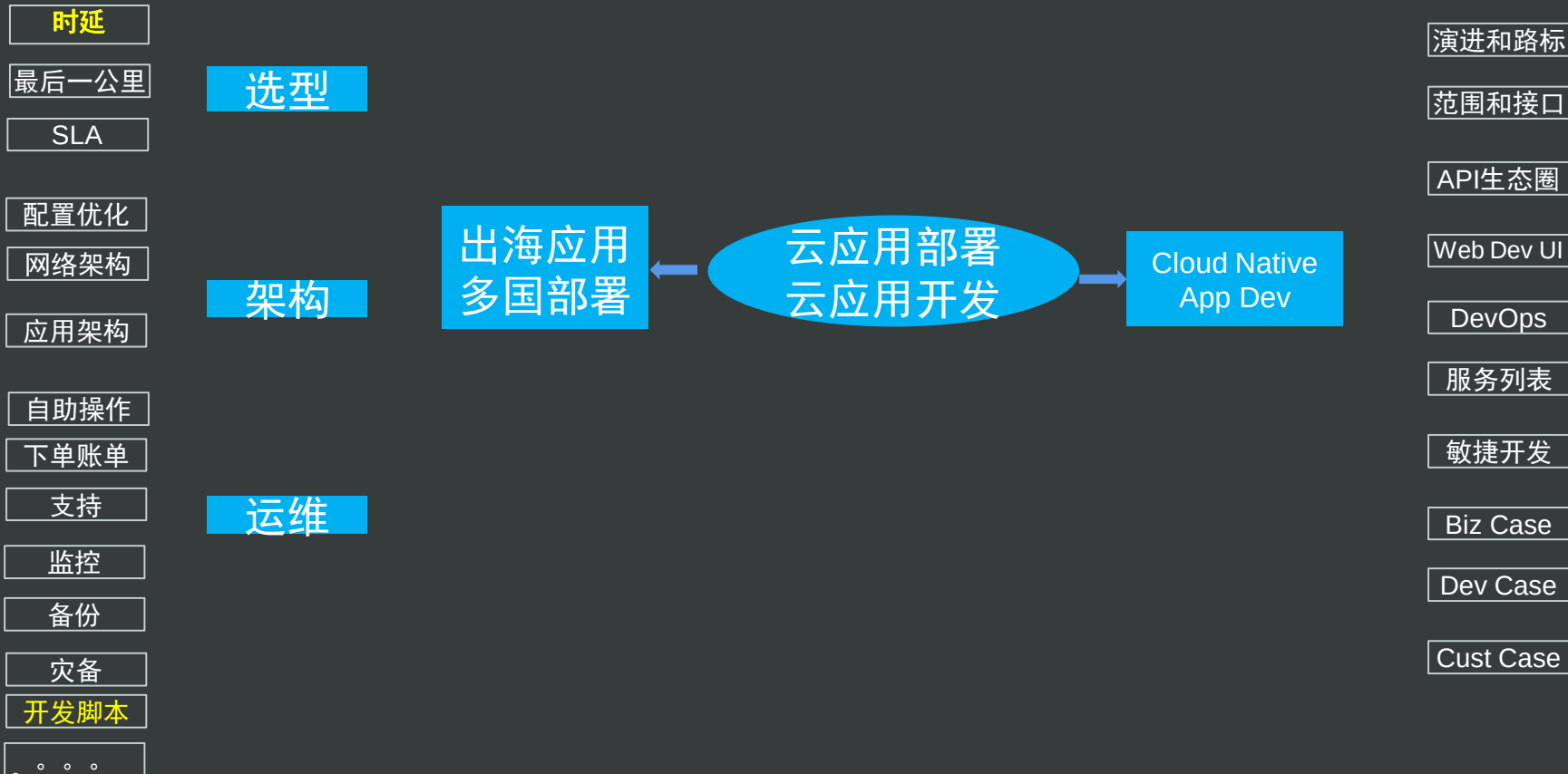
IBM SoftLayer 云上敏捷开发实践 和海外多国部署与灾备

蒋旭春

Agenda



Agenda



我们要在面向俄罗斯用户运营游戏，应该选那个城市的SL DC部署？

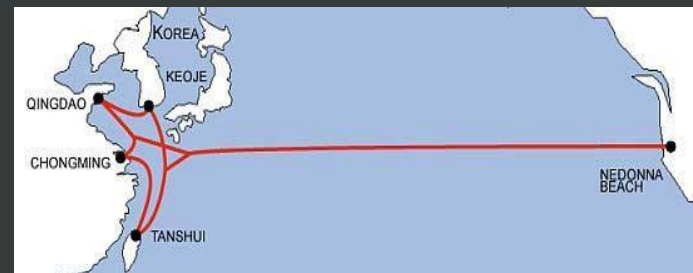
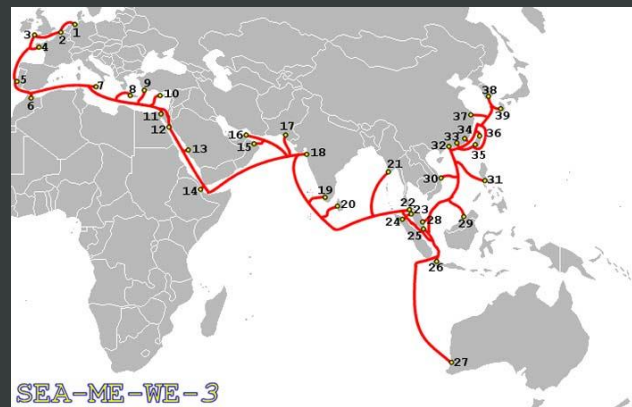
东南亚，印尼，泰国，马来，越南，选新加坡还是香港？

已经在伦敦部署整套(App,DB)，刚赢了曼谷客户，能不能只在香港部署Web，调用伦敦的DB？

巴西用户手机上传视频 UGC 应用，转码在美国，网络体验不好，怎么办？

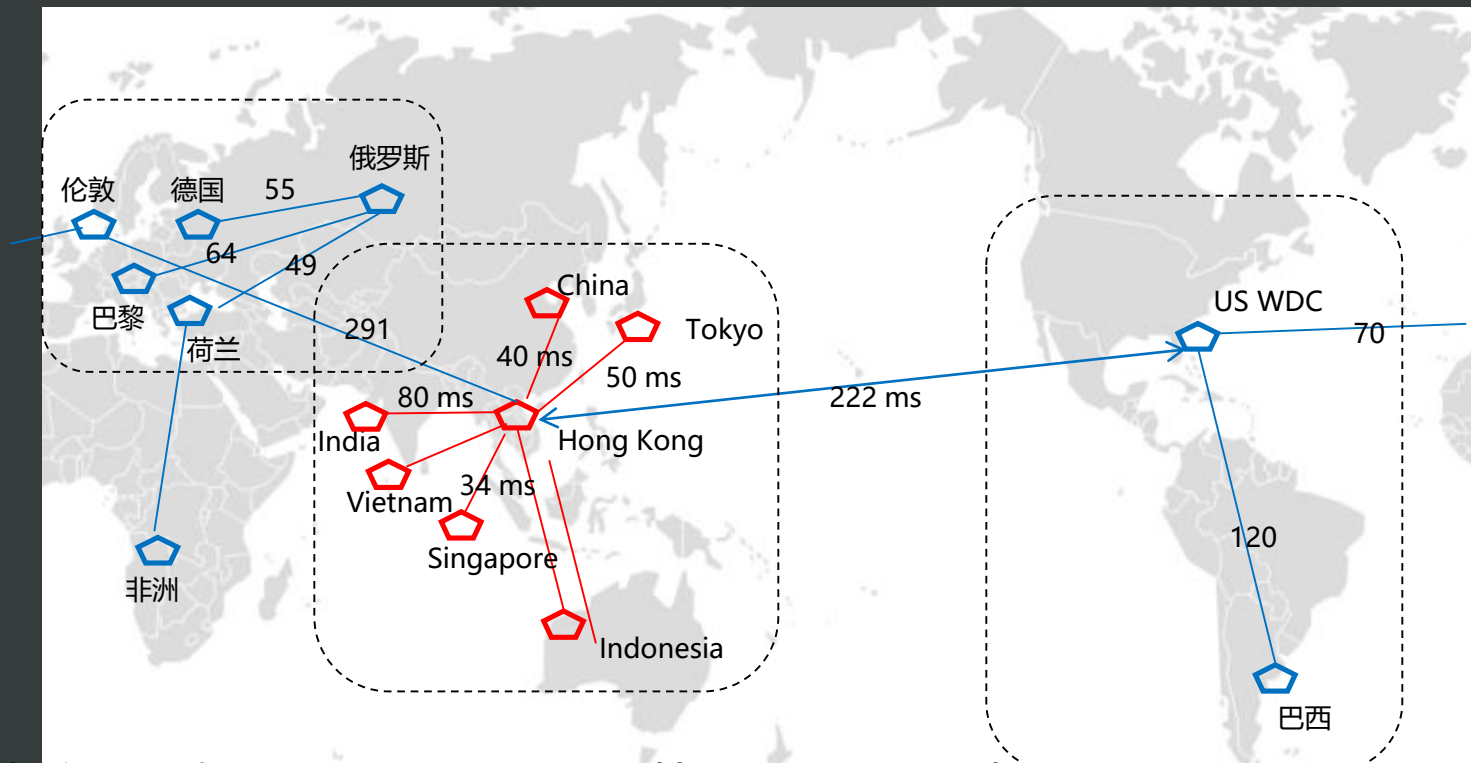
智利，伊朗，伊拉克，怎么办？

全球同服，内网调用，怎么做？



Latency (from various customer projects)

Latency TOK-HK 50ms, SGP-HK 40ms, India-HK 80ms, SH/BJ -HK 40ms, 俄罗斯到荷兰DC快, 巴西到US WDC 快, ...



因为流量批发,BGP优化,CDN加速等原因, 实际测试最好由企业提供当地IP最准确

SoftLayer IP Backbone - Network Latency

[All](#) | [North America](#) | [Europe](#) | [Asia / Pacific](#)

	ATL	CHI	DAL	DEN	HOU	LAX	MEX	MIA	MON	NYC	SEA	SJC	TOR	WDC	AMS	FRA	LON	MIL	PAR	CHE	HKG	MEL	SNG	SYD	TOK	SAO
ATL		31	20	34	28	54	44	13	27	18	61	61	33	13	92	98	85	107	109	233	207	200	239	192	160	124
CHI	29		24	25	28	48	48	40	19	22	43	52	11	19	94	104	89	108	96	212	174	201	194	193	126	136
DAL	20	21		15	6	33	25	30	38	41	41	37	31	31	109	122	107	129	113	247	184	180	216	167	132	141
DEN	34	25	15		22	34	39	43	44	46	27	28	36	42	116	125	114	132	118	220	174	183	205	171	126	154
HOU	28	25	6	22		36	30	25	47	45	46	42	36	37	116	123	113	134	118	252	190	188	220	173	135	137
LAX	54	53	32	34	33		48	60	69	69	25	8	65	64	142	158	145	165	141	218	155	149	186	137	109	187
MEX	44	45	25	39	30	53		54	63	66	65	56	LAX - SJC Latency - 5 Minute Average Current Warning Critical Latency: 8ms 10ms 12ms Loss: 0% 2% 4%							266	203	199	232	186	151	168
MIA	13	41	30	43	24	60	54		39	32	73	65								247	215	210	242	194	176	111
MON	27	19	38	44	47	67	63	39		9	63	64								205	217	212	249	200	144	122
NYC	18	21	41	46	49	69	65	32	9		63	73								209	195	221	225	212	143	114
SEA	73	43	41	27	46	25	65	70	64	61		18	54	58	133	146	133	152	138	183	130	177	155	162	82	179
SJC	61	45	37	26	42	8	56	70	64	66	18		56	60	137	149	133	161	138	201	143	158	163	146	98	180
TOR	31	0	31	20	36	51	55	44	9	15	54	56		21	88	93	82	106	88	217	210	206	241	193	137	126
WDC	13	17	31	39	37	73	56	26	15	7	58	60	21		80	87	73	95	89	212	214	209	247	198	137	117
AMS	94	94	109	111	116	144	134	105	81	71	134	137	88	80		7	6	17	10	136	197	257	171	263	236	187
FRA	98	103	122	125	123	154	144	111	88	83	144	149	94	87	7		16	10	10	133	200	244	170	260	222	187
LON	85	88	107	114	113	145	132	97	76	67	133	133	82	72	6	15		23	8	137	199	253	163	252	210	180
MIL	111	119	129	137	134	160	154	124	101	93	152	159	107	95	18	10	22		17	143	203	256	170	277	243	207
PAR	108	94	113	118	119	141	138	120	82	73	138	138	88	89	10	10	8	17		131	188	240	151	246	239	185
CHE	233	224	247	226	252	218	266	245	205	201	183	201	216	214	136	133	131	141	119		64	116	33	126	99	311
HKG	224	170	184	174	190	155	203	212	217	193	130	144	210	214	197	200	199	203	188	64		115	33	114	47	327
MEL	200	200	180	183	188	149	199	210	212	220	177	158	206	209	257	244	253	256	245	116	115		84	14	126	321
SNG	265	206	217	201	220	184	232	246	249	246	155	163	240	248	172	170	163	173	151	33	33	84		96	67	356
SYD	192	193	167	171	174	137	186	194	200	210	163	146	193	198	263	267	252	277	246	127	114	14	96		113	307
TOK	160	124	131	126	135	109	149	173	145	143	82	98	137	137	241	223	211	241	239	99	47	126	67	113		276
SAO	124	135	141	154	137	187	169	111	122	114	179	180	126	117	185	190	179	205	184	309	327	321	356	307	276	

足不出户，测试海外用户覆盖和机房选址

选址全球任意机房（冗余），目标IP地址，执行 Ping, Traceroute, BGP 测试。对比后选址满意的SL城市节点

SoftLayer IP Backbone :: Looking Glass

Command:

- ☒ Ping
☐ Traceroute
☐ BGP Table Lookup
☐ BGP Summary

Color Scheme:

White / Black

Select Location and Router

Frankfurt, DE - InterXion [bbr01.xn01.fra01]

Target Address/Subnet or Hostname

www.ox.ac.uk

RDNS: aurochs-web-155.nsms.ox.ac.uk

Execute



进行人机身份验证



Looking Glass Results

```
bbr01.xn01.fra01> ping 129.67.242.155
PING 129.67.242.155 (129.67.242.155): 56 data bytes
64 bytes from 129.67.242.155: icmp_seq=0 ttl=54 time=16.792 ms
64 bytes from 129.67.242.155: icmp_seq=1 ttl=54 time=16.663 ms
64 bytes from 129.67.242.155: icmp_seq=2 ttl=54 time=16.619 ms
64 bytes from 129.67.242.155: icmp_seq=3 ttl=54 time=17.464 ms
64 bytes from 129.67.242.155: icmp_seq=4 ttl=54 time=16.814 ms

--- 129.67.242.155 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max/stddev = 16.619/16.870/17.464/0.306 ms
```

Select Location and Router

Frankfurt, DE - InterXion [bbr01.xn01.fra01]

Seattle, WA - Westin Building [bbr02.wb01.sea02]

Toronto, CA - Cologix [bbr01.cl01.tor02]

Toronto, CA - Cologix [bbr02.cl01.tor02]

Europe

Amsterdam, NL - Equinix [bbr01.eq01.ams02]

Amsterdam, NL - Equinix [bbr02.eq01.ams02]

Frankfurt, DE - InterXion [bbr01.xn01.fra01]

Frankfurt, DE - InterXion [bbr02.xn01.fra01]

London, UK - Telecity [bbr01.tg01.lon01]

London, UK - Telecity [bbr02.tg01.lon01]

Milan, IT - Infracom [bbr01.ic01.mil02]

Milan, IT - Infracom [bbr02.ic01.mil02]

Oslo, NO - Verizon [bbr01.vz01.osl02]

Oslo, NO - Verizon [bbr02.vz01.osl02]

Paris, FR - Equinix [bbr01.eq01.par02]

Paris, FR - Equinix [bbr02.eq01.par02]

Asia

Chennai, IN [bbr01.sr01.che01]

Chennai, IN [bbr02.sr01.che01]

Hong Kong, HK - Mega-I [bbr01.pn01.hkg01]

Tips and Steps

OSC 源创会

Opensource Innovation Meetup

IT大咖说

<http://www.qkankan.com/>

<http://www.world68.com/sort.asp?bid=96> 印尼

<http://www.dragon-guide.net/guobie/feizhou/Egypt.htm> 埃及

Steps:

1. 找到多个当地用户国家的IP， 多选几个防止干扰。（例如俄罗斯运营商自己的网站IP必然很快， 普通公司网站IP是正常速度）
2. 验证确认该给geo IP物理上属于当地国家， 不是在其他国家托管的， 否者测试结果不是这个国家的， 没有意义。
3. 在lg.softlayer.com 点击 I agree, 选择SL某个城市地点的测试网关， 对上面测试IP执行测试
4. 分析结果， 记下同一个用户地址所在地IP(例如俄罗斯)， 用不同城市(例如荷兰， 挪威， 德国)的时延对比， 选择最快节点。

<http://ip.chinaz.com/ipbatch>

<http://ip.chinaz.com/siteip>

<http://whatismyipaddress.com/ip-lookup>

<https://www.iplocation.net/>

<http://ipinfo.io/>

[http:// www.geoiptool.com](http://www.geoiptool.com)

<http://www.ip2location.com/>

<http://www.ip-tracker.org/locator/ip-lookup.php>

http://www.ip-adress.com/ip_tracer

<http://lg.softlayer.com/>

<http://cloudmonitor.ca.com/en/ping.php>

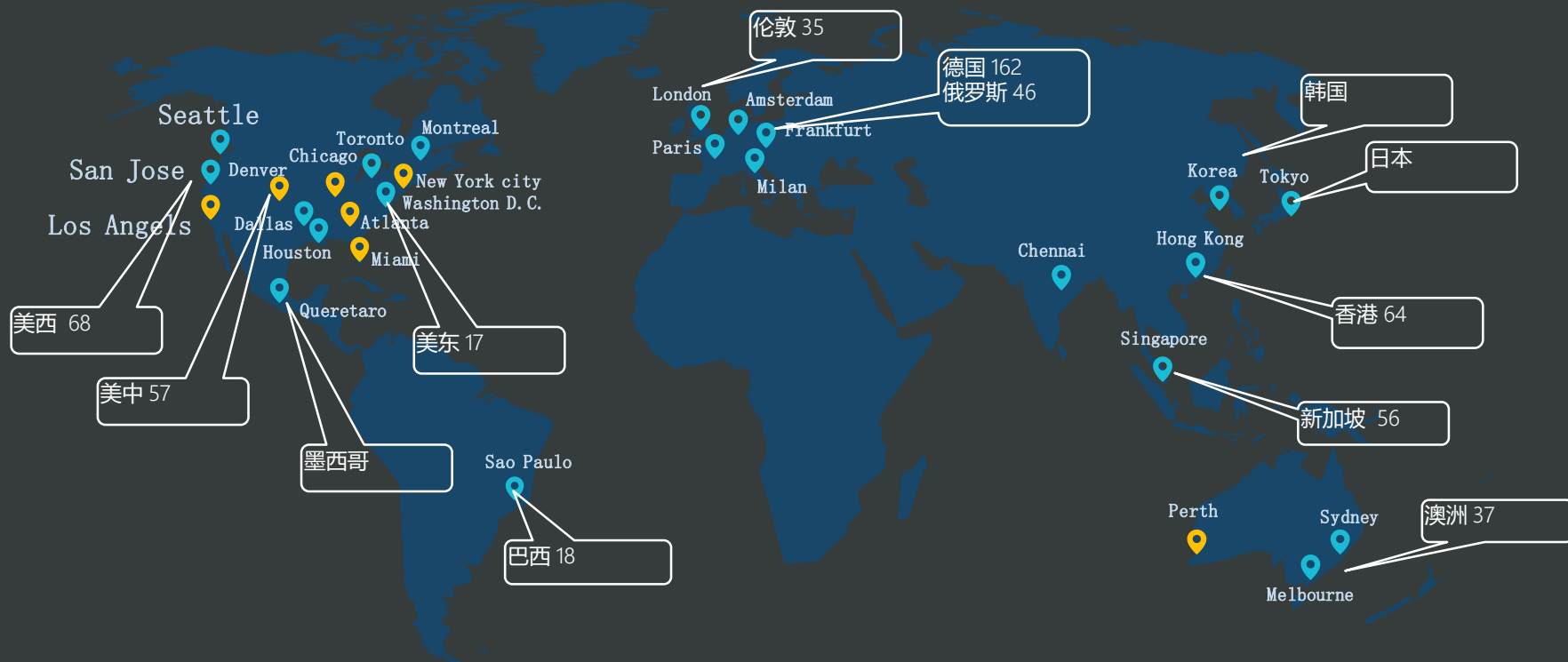
<http://tool.l7mon.cn/traceroute.php>

<http://looking-glass.telialia.net/>

<http://www.speedtest.net>

*真实用户地址群发起的测试服务， 覆盖本地运营商

Case 1

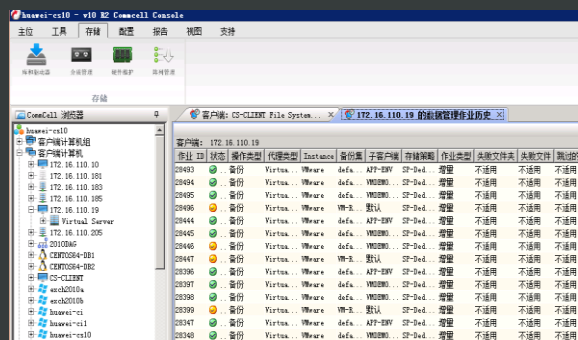
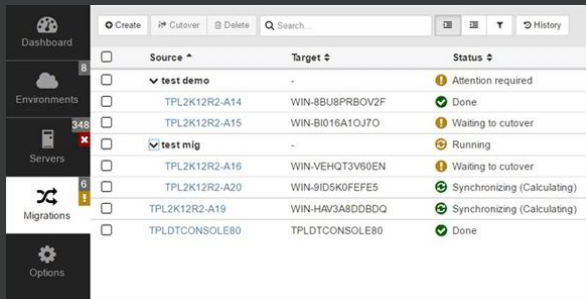


备份和灾备

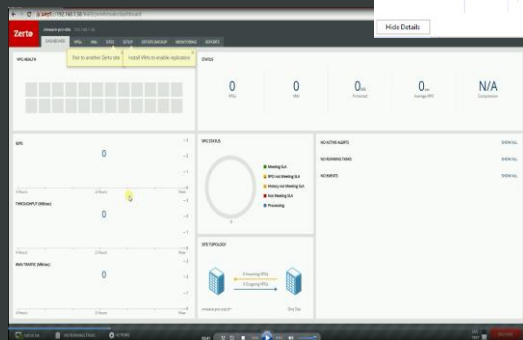
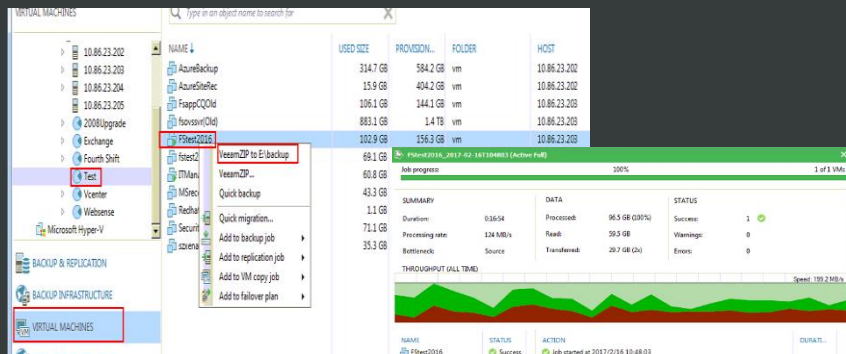
Image

- Reboot
- Power On/Off
- Rescue
- OS Reload
- Load From Image
- Boot From Image
- Create Image Template
- Create Flex Image
- Migrate to Local Disk
- Rename instance
- Port Control
- View Audit Logs
- KVM Console
- Cancel Device

eVault, commvault

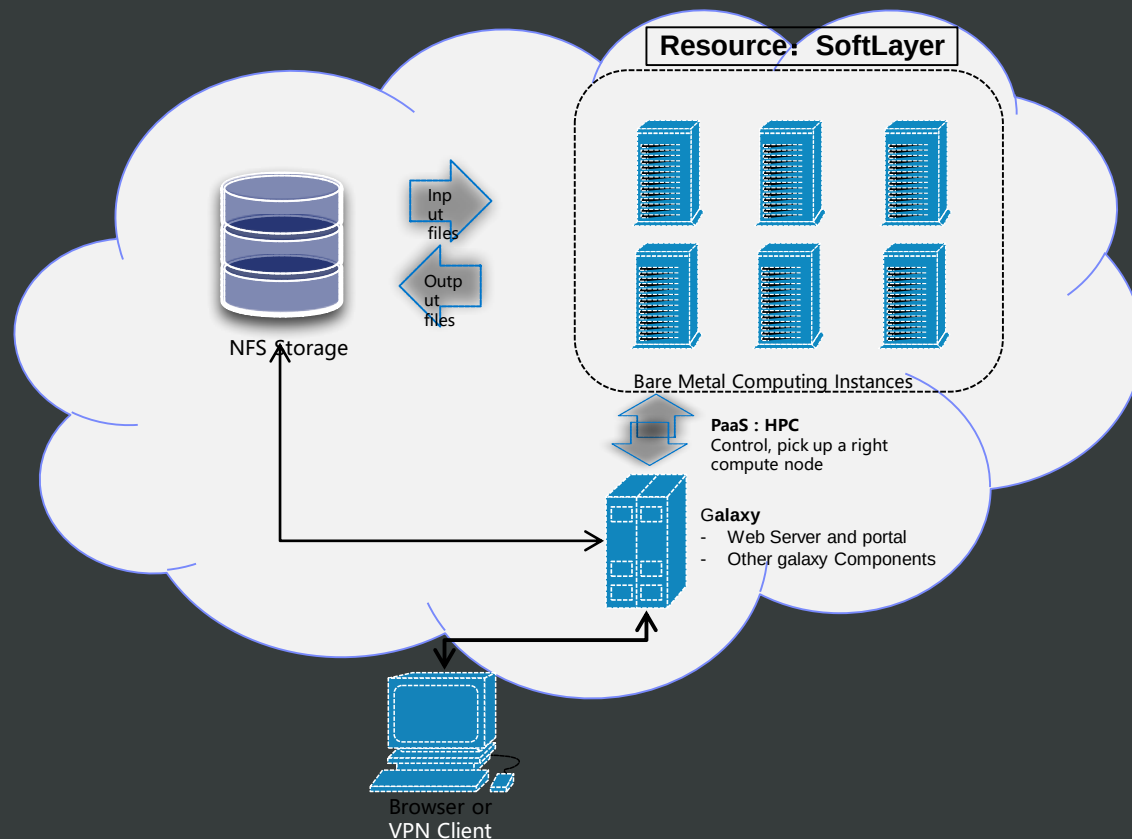


R1Soft, Veeam, Zerto



脚本 Auto Scale

基因生物分析应用的逻辑结构

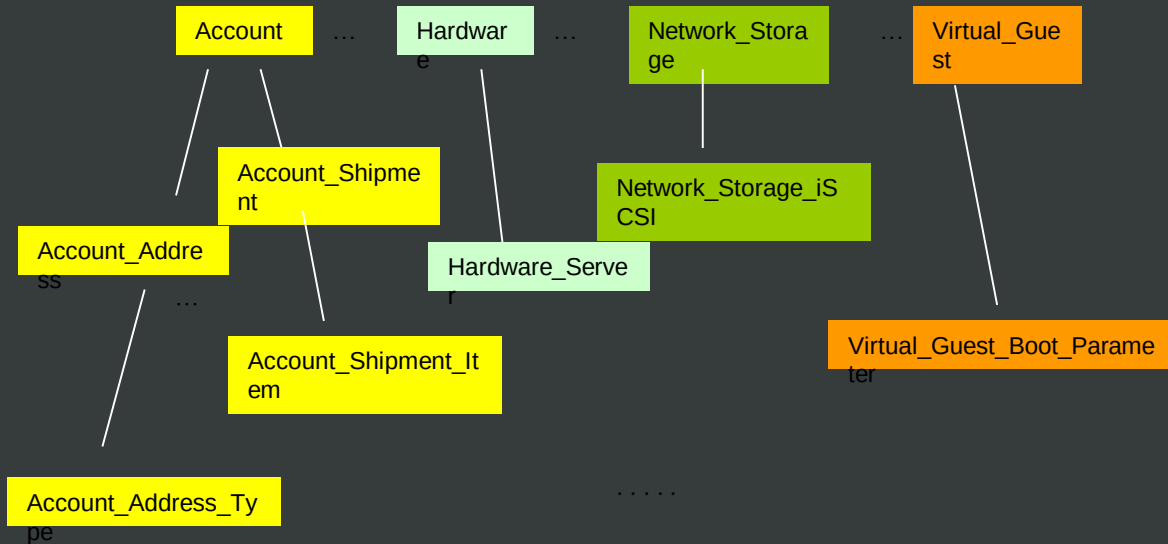


Execution Sequence (Possibly)

- User submit a request to the web portal
- User upload a file-to-be analysis or specify a file in the cloud (NFS)
- Scheduler pick up a appropriate compute node for the request
- Computing node get the file and do the compute
- Computing node put the result file
- Notify user all is done
- User get the file or view the result online

脚本 API

APIs – services and data structures hierarchy



Example API service categories include

- User Accounts
- User Billing
- CCI Management (CloudLayer Computing Instance – Virtual Guest)
- Hardware management (dedicated/bare metal and other hardware)
- Product Ordering
- Configuration Templates
- Software Components
- Locations
- Network (firewalls, gateways, load balancers, subnets, and VLANs)
- Storage (iSCSI, NAS, and backup)
- Reboots and reloads
- Ticketing
- DNS
- Security (certificates, keys, and scans)
- Monitoring
- Portal customization
- Auxiliary

SoftLayer API at a glance

—SoftLayer [main] API

- Version 3.0 API provides 2,200 function calls to over 180 services
- Supports REST, SOAP & XML-RPC interfaces
- Clients implemented in C#, Perl, PHP, Python, Ruby, VB .Net, command line Python client
- Is a major direct source of revenue
- Accessible from mobile devices as well

Object Store API:

- Based on OpenStack Swift
- REST-based
- Clients: Java, Ruby, PHP, Python
- Supports Containers and Objects
- Content Delivery Network integration
- Search integration

Message Queue API:

- REST-based
- 64k max
- Key/value pair message fields
- Clients: C#, Java, Ruby, PHP, Python
- Supports Topics, Topic Tags, Subscriptions
- Persistent via Cloudant data layer

Basic Concepts:

- Endpoint (public or private)
- Service (e.g. Account, Storage..)
- Method or Action (e.g. getUsers())
- Data Types (String, boolean, int and Complex)
- Properties (local, relational, count)
- Result limits (rows filtering)
- Object Masks (properties filtering)

To begin using the API:

- Enable the API use from Customer Portal
- Create unique secret API key
- Authentication based on user and API key
- Choose language client library

脚本 API

API CLI (Command Line Interface)

```
File Edit View Search Terminal Help
(slapi)[dima@oc8715403615 objectstore]$ sl --help
usage: sl <module> [<args>...]
       sl help <module>
       sl help <module> <command>
       sl [-h | --help]

SoftLayer Command-line Client

The available modules are:
ccci      Manage, delete, order compute instances
config    View and edit configuration for this tool
dns       Manage DNS
firewall  Firewall rule and security management
hardware  View hardware details
bmetal    Interact with bare metal instances
help      Show help
iscsi     View iSCSI details
image     Manages compute and flex images
metadata  Get details about this machine. Also available with 'my' and 'meta'
nas       View NAS details
ssl       Manages SSL

See 'sl help <module>' for more information on a specific module.

To use most commands your SoftLayer username and api_key need to be configured.
The easiest way to do that is to use: 'sl config setup'
(slapi)[dima@oc8715403615 objectstore]$
```

List VM

```
File Edit View Search Terminal Help
(slapi)[dima@oc8715403615 objectstore]$ sl help cci list
usage: sl cci list [--hourly | --monthly] [--sortby=SORT_COLUMN] [--tags=TAGS]
[options]

List CCIs

Examples:
sl cci list --datacenter=dal05
sl cci list --network=100 --cpu=2
sl cci list --memory='>= 2048'
sl cci list --tags=production,db

Options:
--sortby=ARG      Column to sort by. options: id, datacenter, host,
                  Cores, memory, primary_ip, backend_ip

Filters:
--hourly          Show hourly instances
--monthly         Show monthly instances
-H --hostname=HOST Host portion of the FQDN. example: server
-D --domain=DOMAIN Domain portion of the FQDN. example: example.com
-c --cpu=CPU      Number of CPU cores
-m --memory=MEMORY Memory in mebibytes (n * 1024)
-d DC, --datacenter=DC datacenter shorthand (sng01, dal05, ...)
-n MBPS, --network=MBPS Network port speed in Mbps
--tags=ARG        Only show instances that have one of these tags.
                  Comma-separated. (production,db)

For more on filters see 'sl help filters'

Standard Options:
--format=ARG      Output format. [Options: table, raw] [Default: table]
-C FILE --config=FILE Config file location. [Default: ~/.softlayer]
--debug=LEVEL     Specifies the debug noise level
                  1=warn, 2=info, 3=debug
-h --help         Show this screen
(slapi)[dima@oc8715403615 objectstore]$

File Edit View Search Terminal Help
(slapi)[dima@oc8715403615 objectstore]$ sl cci list --datacenter=ams01 --format=raw
1903553 ams01 amsterdam.dc5.jeffsloyer.com 2 4096 37.58.87.82 10.68.62.130 NULL
1905013 ams01 jsenams.softlayer.com 1 1024 37.58.75.133 10.68.62.138 NULL
1905011 ams01 jsvyams.softlayer.com 1 1024 37.58.75.132 10.68.62.131 NULL
2001941 ams01 miravirt-01.softlayer.com 1 2048 37.58.87.84 10.68.62.134 NULL
1964570 ams01 tm1pocrdp.softlayer.com 8 16384 5.10.80.243 10.68.62.148 NULL
(slapi)[dima@oc8715403615 objectstore]$ sl cci list --datacenter=ams01 --format=table
.....
: id : datacenter : host : cores : memory : primary_ip : backend_ip : active_transaction :
.....
: 1903553 : ams01 : amsterdam.dc5.jeffsloyer.com : 2 : 4G : 37.58.87.82 : 10.68.62.130 : - :
: 1905013 : ams01 : jsenams.softlayer.com : 1 : 1G : 37.58.75.133 : 10.68.62.138 : - :
: 1905011 : ams01 : jsvyams.softlayer.com : 1 : 1G : 37.58.75.132 : 10.68.62.131 : - :
: 2001941 : ams01 : miravirt-01.softlayer.com : 1 : 2G : 37.58.87.84 : 10.68.62.134 : - :
: 1964570 : ams01 : tm1pocrdp.softlayer.com : 8 : 16G : 5.10.80.243 : 10.68.62.148 : - :
.....
(slapi)[dima@oc8715403615 objectstore]$
```


脚本 API

API CLI (Command Line Interface)

```
File Edit View Search Terminal Help
(slapi)[dima@oc8715403615 objectstore]$ sl --help
usage: sl <module> [<args>...]
       sl help <module>
       sl help <module> <command>
       sl [-h | --help]

SoftLayer Command-line Client

The available modules are:
  cci          Manage, delete, order compute instances
  config       View and edit configuration for this tool
  dns          Manage DNS
  firewall     Firewall rule and security management
  hardware     View hardware details
  bmetal       Interact with bare metal instances
  help         Show help
  iscsi        View iSCSI details
  image        Manages compute and flex images
  metadata     Get details about this machine. Also available with 'my' and 'meta'
  nas          View NAS details
  ssl          Manages SSL

See 'sl help <module>' for more information on a specific module.

To use most commands your SoftLayer username and api_key need to be configured.
The easiest way to do that is to use: 'sl config setup'
(slapi)[dima@oc8715403615 objectstore]$
```

List VM

```
File Edit View Search Terminal Help
(slapi)[dima@oc8715403615 objectstore]$ sl help cci list
usage: sl cci list [--hourly | --monthly] [--sortby=SORT_COLUMN] [--tags=TAGS]
[options]

List CCIs

Examples:
  sl cci list --datacenter=dal05
  sl cci list --network=100 --cpu=2
  sl cci list --memory='>= 2048'
  sl cci list --tags=production,db

Options:
  --sortby=ARG      Column to sort by. options: id, datacenter, host,
                    Cores, memory, primary_ip, backend_ip

Filters:
  --hourly          Show hourly instances
  --monthly         Show monthly instances
  -H --hostname=HOST Host portion of the FQDN. example: server
  -D --domain=DOMAIN Domain portion of the FQDN. example: example.com
  -c --cpu=CPU       Number of CPU cores
  -m --memory=MEMORY Memory in mebibytes (n * 1024)
  -d DC, --datacenter=DC datacenter shorthand (sng01, dal05, ...)
  -n MBPS, --network=MBPS Network port speed in Mbps
  --tags=ARG        Only show instances that have one of these tags.
                    Comma-separated. (production,db)

For more on filters see 'sl help filters'

Standard Options:
  --format=ARG      Output format. [Options: table, raw] [Default: table]
  -C FILE --config=FILE Config file location. [Default: ~/.softlayer]
  --debug=LEVEL     Specifies the debug noise level
                    1=warn, 2=info, 3=debug
  -h --help         Show this screen
(slapi)[dima@oc8715403615 objectstore]$

File Edit View Search Terminal Help
(slapi)[dima@oc8715403615 objectstore]$ sl cci list --datacenter=ams01 --format=raw
1903553 ams01 amsterdam.dc5.jeffsloyer.com 2 4096 37.58.87.82 10.68.62.130 NULL
1905013 ams01 jsenams.softlayer.com 1 1024 37.58.75.133 10.68.62.138 NULL
1905011 ams01 jsvyams.softlayer.com 1 1024 37.58.75.132 10.68.62.131 NULL
2001941 ams01 miravirt-01.softlayer.com 1 2048 37.58.87.84 10.68.62.134 NULL
1964570 ams01 tm1pocrdp.softlayer.com 8 16384 5.10.80.243 10.68.62.148 NULL
(slapi)[dima@oc8715403615 objectstore]$ sl cci list --datacenter=ams01 --format=table
.....
id : datacenter : host : cores : memory : primary_ip : backend_ip : active_transaction :
.....
1903553 : ams01 : amsterdam.dc5.jeffsloyer.com : 2 : 4G : 37.58.87.82 : 10.68.62.130 : - :
1905013 : ams01 : jsenams.softlayer.com : 1 : 1G : 37.58.75.133 : 10.68.62.138 : - :
1905011 : ams01 : jsvyams.softlayer.com : 1 : 1G : 37.58.75.132 : 10.68.62.131 : - :
2001941 : ams01 : miravirt-01.softlayer.com : 1 : 2G : 37.58.87.84 : 10.68.62.134 : - :
1964570 : ams01 : tm1pocrdp.softlayer.com : 8 : 16G : 5.10.80.243 : 10.68.62.148 : - :
(slapi)[dima@oc8715403615 objectstore]$
```

脚本 API

API Create Bare Metal Server

```
File Edit View Search Terminal Help
(slapi)[dima@oc8715403615 objectstore]$ sl bmetal create --hostname=lobo --domain=softlayer.com --cpu=2 --os=UBUNTU_12_64_MINIMAL --disk=250 --memory=2 --hourly
This action will incur charges on your account. Continue? [y/N]: N
Aborting bare metal instance order.
(slapi)[dima@oc8715403615 objectstore]$ sl bmetal create --hostname=lobo --domain=softlayer.com --cpu=2 --os=UBUNTU_12_64_MINIMAL --disk=250 --memory=2 --hourly --test
.....
Item : cost :
.....
2 x 2.0 GHz Core Bare Metal Instance - 2 GB Ram : 0.42 :
250GB SATA II : 0.00 :
Ubuntu Linux 12.04 LTS Precise Pangolin - Minimal Install (64 bit) : 0.00 :
10 Mbps Public & Private Networks : 0.00 :
1 IP Address : 0.00 :
0 GB Bandwidth : 0.00 :
Host Ping : 0.00 :
Nessus Vulnerability Assessment & Reporting : 0.00 :
Automated Notification : 0.00 :
Unlimited SSL VPN Users & 1 PPTP VPN User per account : 0.00 :
Reboot / Remote Console : 0.00 :
Email and Ticket : 0.00 :
Total hourly cost : 0.42 :
.....
-- ! Prices reflected here are retail and do not take account level discounts and are not guaranteed.
(slapi)[dima@oc8715403615 objectstore]$ sl bmetal create --hostname=lobo --domain=softlayer.com --cpu=2 --os=UBUNTU_12_64_MINIMAL --disk=250 --memory=2 --hourly --really
.....
name : value :
.....
id : 1375046 :
created : 2013-07-21T16:18:10-07:00 :
(slapi)[dima@oc8715403615 objectstore]$ sl cci detail 1375046
```

Listing storage and firewalls

```
File Edit View Search Terminal Help
(slapi)[dima@oc8715403615 code]$ sl iscsi list
.....
id : datacenter : size : username : password : server :
.....
1667215 : dal01 : 20GB : SLI257799-1 : ZK9THW28d5C : 10.0.81.23 :
1761104 : sjc01 : 40GB : SLI257799-2 : FNL4JZVM3USL : 10.1.197.21 :
1857872 : sng01 : 20GB : SLI257799-3 : YAXn6vS2KgKN : 10.2.37.21 :
1861846 : sjc01 : 20GB : SLI257799-4 : K8MaHvKRAmgV : 10.1.197.21 :
1867902 : sng01 : 20GB : SLI257799-5 : TZHNZSKTarp9 : 10.2.37.21 :
1868676 : sng01 : 20GB : SLI257799-6 : EtbeqT7zc7vk : 10.2.37.21 :
1869703 : sng01 : 20GB : SLI257799-7 : UvFY9c73eV6J : 10.2.37.21 :
1883813 : wdc01 : 20GB : SLI257799-8 : LQZE16KYJwnS : 10.1.113.10 :
1913332 : dal01 : 20GB : SLI257799-9 : RY69MQ9cJvr6 : 10.0.81.23 :
1913333 : dal01 : 20GB : SLI257799-10 : S313c6Y9nrKj : 10.0.81.23 :
1928915 : wdc01 : 20GB : SLI257799-11 : DFNjCNde7td2 : 10.1.113.10 :
1944339 : dal01 : 100GB : SLI257799-12 : KrcbXbZUGG62 : 10.0.81.23 :
.....
(slapi)[dima@oc8715403615 code]$ sl help nas
usage: sl nas [<command>] [<args>...] [options]

Manage NAS accounts

The available commands are:
list List NAS accounts
(slapi)[dima@oc8715403615 code]$ sl nas list
.....
id : datacenter : size : username : password : server :
.....
1645790 : sjc01 : 20GB : SL257799-1 : D24XEN9u : nas501.service.softlayer.com :
1648021 : sjc01 : 20GB : SL257799-2 : U7KqHgbd : nas501.service.softlayer.com :
1648066 : sjc01 : 20GB : SL257799-3 : HAd5HDqX : nas501.service.softlayer.com :
1653873 : sjc01 : 100GB : SL257799-4 : Q6pFSZvX : nas501.service.softlayer.com :
1661301 : sjc01 : 500GB : SL257799-5 : XNFjpx2X : nas501.service.softlayer.com :
1662394 : sjc01 : 100GB : SL257799-6 : M33p1M33p : nas501.service.softlayer.com :
1662455 : sjc01 : 100GB : SL257799-7 : M33p0M33p : nas501.service.softlayer.com :
1662456 : sjc01 : 100GB : SL257799-8 : M33p0M33p : nas501.service.softlayer.com :
1663738 : wdc01 : 250GB : SL257799-9 : QjGr2Y2Z7 : nas301.service.softlayer.com :
1663739 : dal05 : 250GB : SL257799-10 : SzwER8FR : nas151.service.softlayer.com :
1761057 : sjc01 : 40GB : SL257799-11 : N77knRWX : nas501.service.softlayer.com :
1831708 : sjc01 : 500GB : SL257799-12 : L7nudjKx : nas501.service.softlayer.com :
1871862 : sng01 : 20GB : SL257799-13 : Qcfju9Pf : nassng0101.service.softlayer.com :
1871988 : dal05 : 20GB : SL257799-14 : V69CvqFW : nas151.service.softlayer.com :
1882032 : wdc01 : 20GB : SL257799-15 : VzDP5U7L : nas301.service.softlayer.com :
1884487 : dal05 : 100GB : SL257799-16 : YFPv4Y3n : nas151.service.softlayer.com :
1909022 : dal05 : 40GB : SL257799-18 : TVvtDr9Q : nas151.service.softlayer.com :
1909634 : sjc01 : 20GB : SL257799-19 : HMa7xLwp : nas501.service.softlayer.com :
```

API List VM, BM, Tickets

```
import SoftLayer.API

username = 'your user name'
api_key = 'your API key'

#Create a client to the SoftLayer_Account API service.
client = SoftLayer.Client('SoftLayer_Account', None, username=
api_username,api_key=api_key)

#Get list of CCIs
server_list = client.getVirtualGuests()

for server in server_list:
    print "id: " + str(server['id']) + " hostname: " + server['hostname'] + "." + server['domain']

#Get list of baremetal instances
server_list = client.getBareMetalInstances()

for server in server_list:
    print "id: " + str(server['id']) + " hostname: " + server['hostname'] + "." + server['domain']

# get the dedicated servers
server_list = client.getHardware()

for server in server_list:
    print "id: " + str(server['id']) + " hostname: " + "hostname: " + server['hostname'] + "." +
server['domain']

# get the list of tickets
tickets = client.getTickets()
```

Listing storage and firewalls

```
import SoftLayer.API

username = 'your user name'
api_key = 'your API key'

#Create a client to the SoftLayer_Account API service.
client = SoftLayer.Client('Account', None, username=
api_username,api_key=api_key)

client.set_result_limit(5)
tickets = client.getTickets()

for ticket in tickets:
    print " ticket id: " + str(ticket['id']) + " title: " + ticket['title']

object_mask = {
    'hardware' : {
        'operatingSystem' : {
            'passwords' : {},
        },
        'networkComponents' : {},
        'datacenter' : {},
        'processorCount' : {},
    }
}

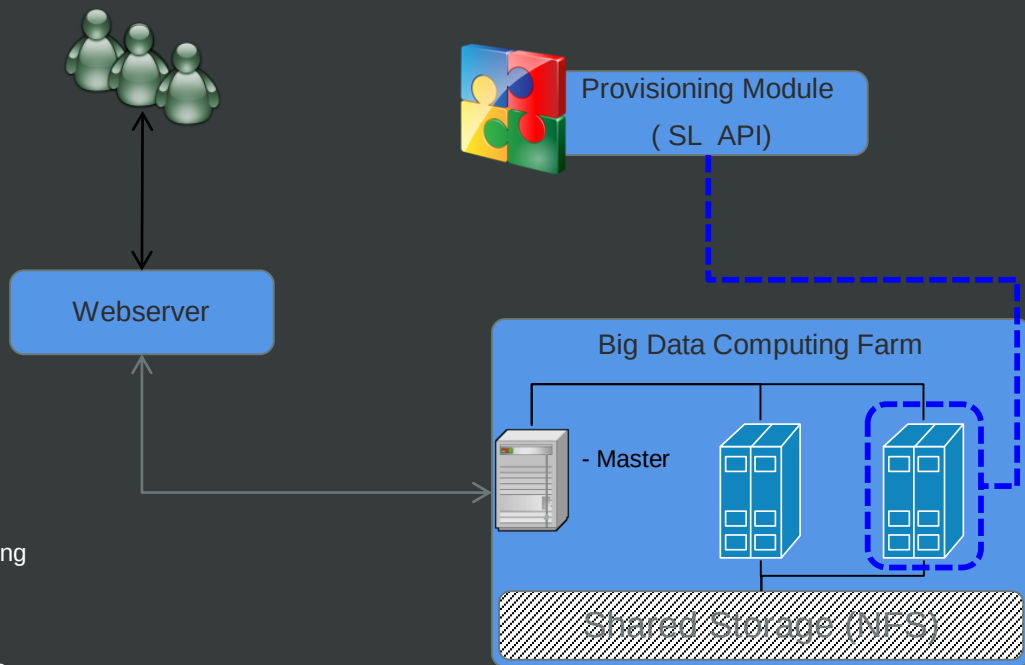
client.set_object_mask(object_mask)

hardware = client.getHardware()

pprint.pprint(hardware)
```

脚本 Auto Scale

基因生物分析应用的逻辑结构



- Create OS image from existing node
- CMD window to execute provisioning API script
- Run post-installation script to configure the SGE cluster
- Run jobs

脚本 Auto Scale

```
import SoftLayer
username='abc'
api_key='b59a7866f7275bb18e3c4d4fe498d8017c3fd200ce20d1189e03096130fe5dc1'

client = SoftLayer.Client(username,api_key)

# list the virtual server.
def getVirtualServerList():
    print "Virtual server list:"
    account = client['Account']
    servers_list = account.getVirtualGuests()
    for server in servers_list:
        print("ID:"+str(server['id'])+" Hostname:"+server['hostname']+" Domain:"+server['domain'])
def getBaremetalServerList():
    print "baremetal server list"
    account = client['Account']
    servers_list = account.getHardware()
    for server in servers_list:
        print("ID:"+str(server['id'])+" Hostname:"+server['hostname']+" Domain:"+server['domain'])

def createStandradImage(instanceID):
    vs = client['Virtual_Guest']
    try:
        vs.createArchiveTransaction("Test Group", [{id: block_devices[0]['id']}], "test the template",id=instanceID)
        print "image create successful!"
    except:
        print "something error"

def createFlexImage(instanceID):
    hws = client['SoftLayer_Hardware_Server']

    #this is a captureImage.
    try:
        result = hws.captureImage(id= 197394)
        print result
    except:
        print 'something error'

def getImageTemplateId():
    templates = client['Account'].getBlockDeviceTemplateGroups()
    for template in templates:
        print template

if __name__ == "__main__":
    getVirtualServerList()

    id_number = int(raw_input("please input the vm instance id you want to create: "))
```

```
"""
function : vm provision script in softlayer.
the vm config can be changed:4cores/32GBMem/100GBHD/Cent OS 6.0
datacenter : Singapore01
"""

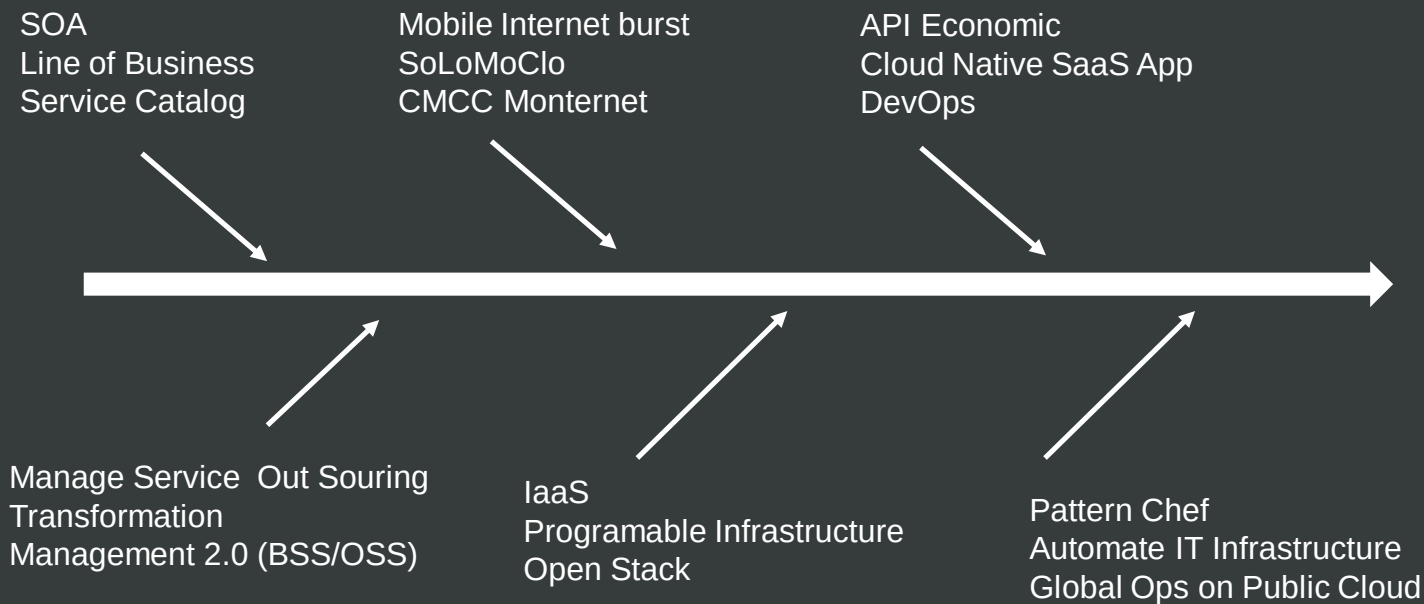
import SoftLayer
username='abc'
api_key='b59a7866f7275bb18e3c4d4fe498d8017c3fd200ce20d1189e03096130fe5dc1'

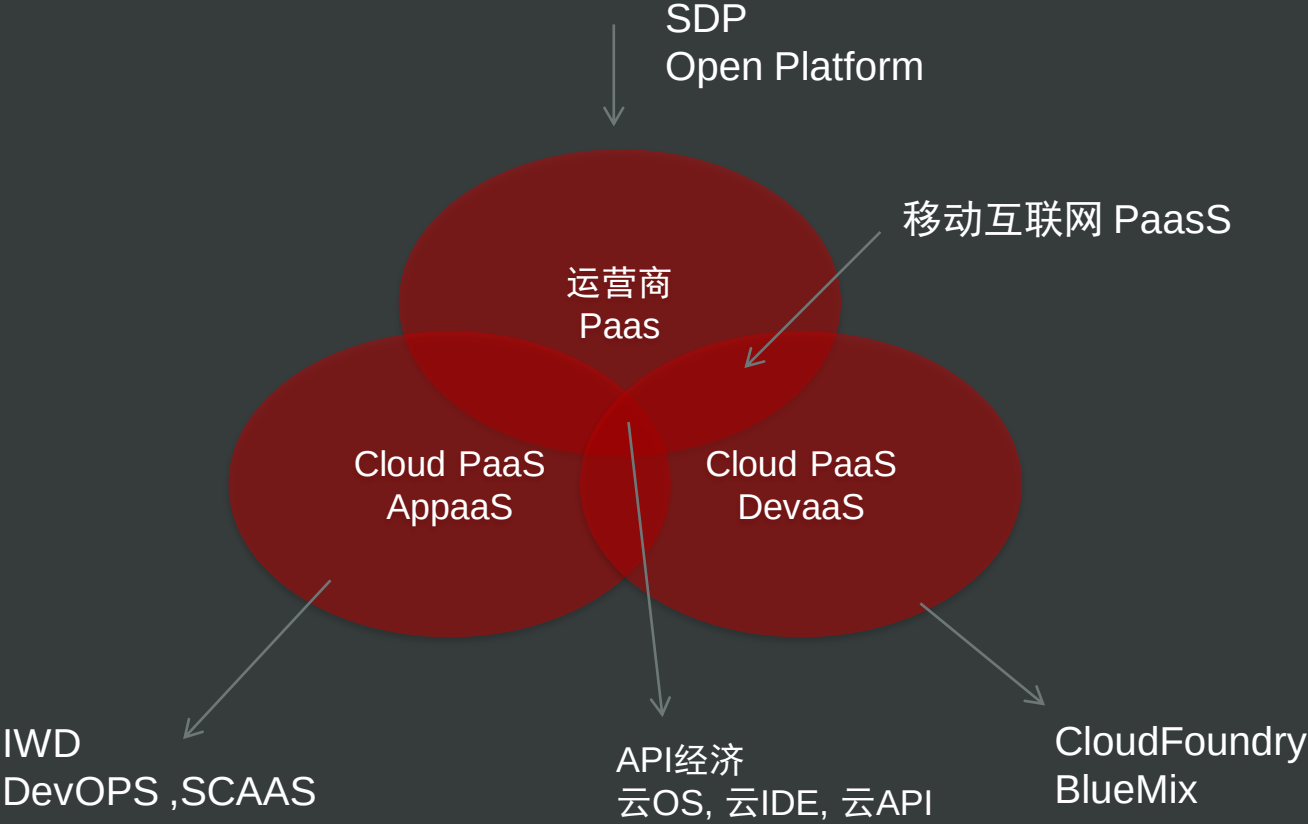
client = SoftLayer.Client(username,api_key)
def submitVirtualServerOrder():
    vs = client['Virtual_Guest']
    newCCI = {
        'hostname': 'bgi',
        'domain': 'ibm.com',
        'startCpus': 4,
        'maxMemory': 32768,
        'hourlyBillingFlag': 'true',
        'operatingSystemReferenceCode': 'CENTOS_6_64',
        'datacenter': {'name':"sng01"},
        'localDiskFlag': 'True',
        'blockDevices': [
            {
                "device": "0",
                "diskImage": {"capacity": 100}}
        ]
    }
    result = vs.createObject(newCCI)
    print result
try:
    submitVirtualServerOrder()
    print "your order submit successful, it may be take several minutes to
    provision."
except:
    print "something error, please check it."
print "-----virtual Server list-----"
account = client['Account']
servers_list = account.getVirtualGuests()
for server in servers_list:
    print("ID:"+str(server['id'])+" Hostname:"+server['hostname']+" Domain:"+server['domain'])
```

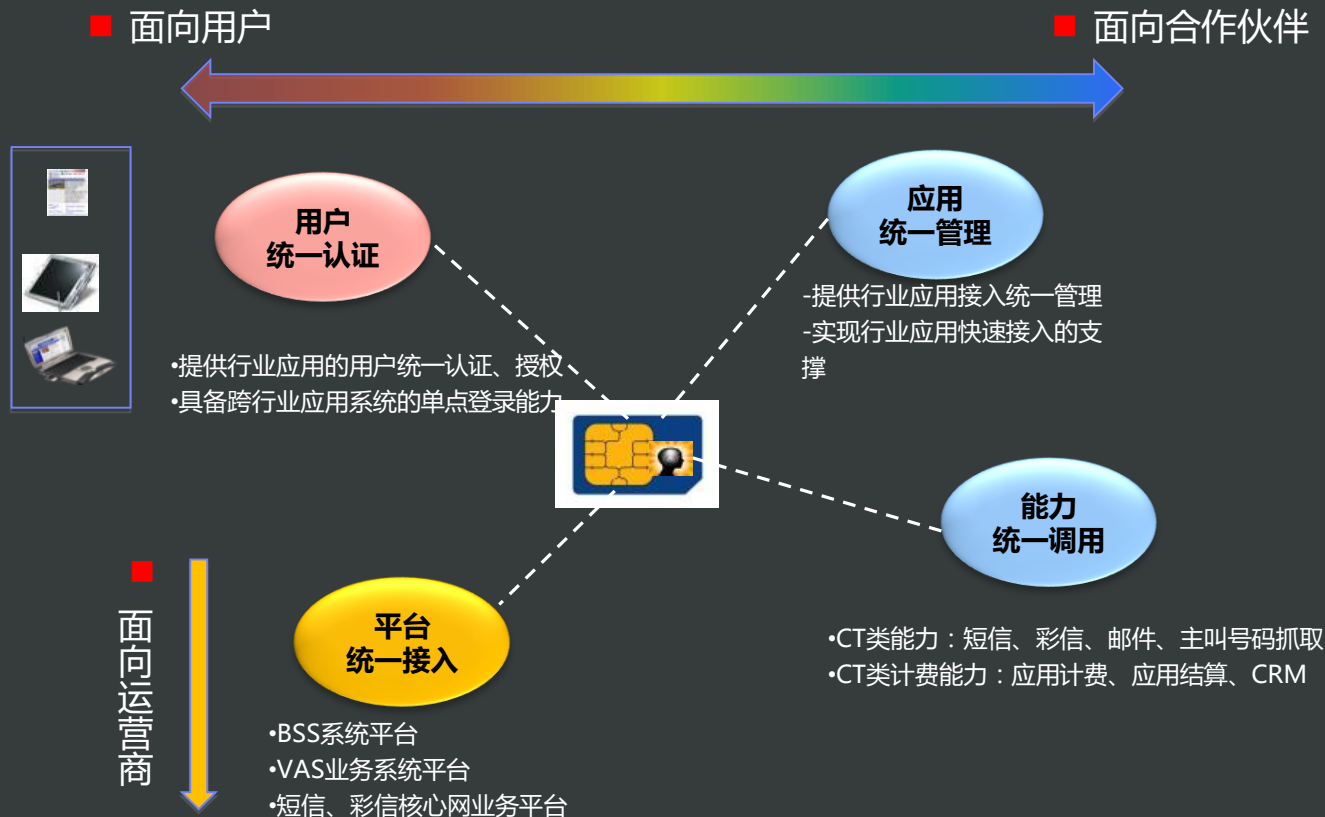
PaaS的历史演进，代表公司，未来路标

OSC 源创会
Opensource Innovation Meetup

IT大咖说
IT 大咖说

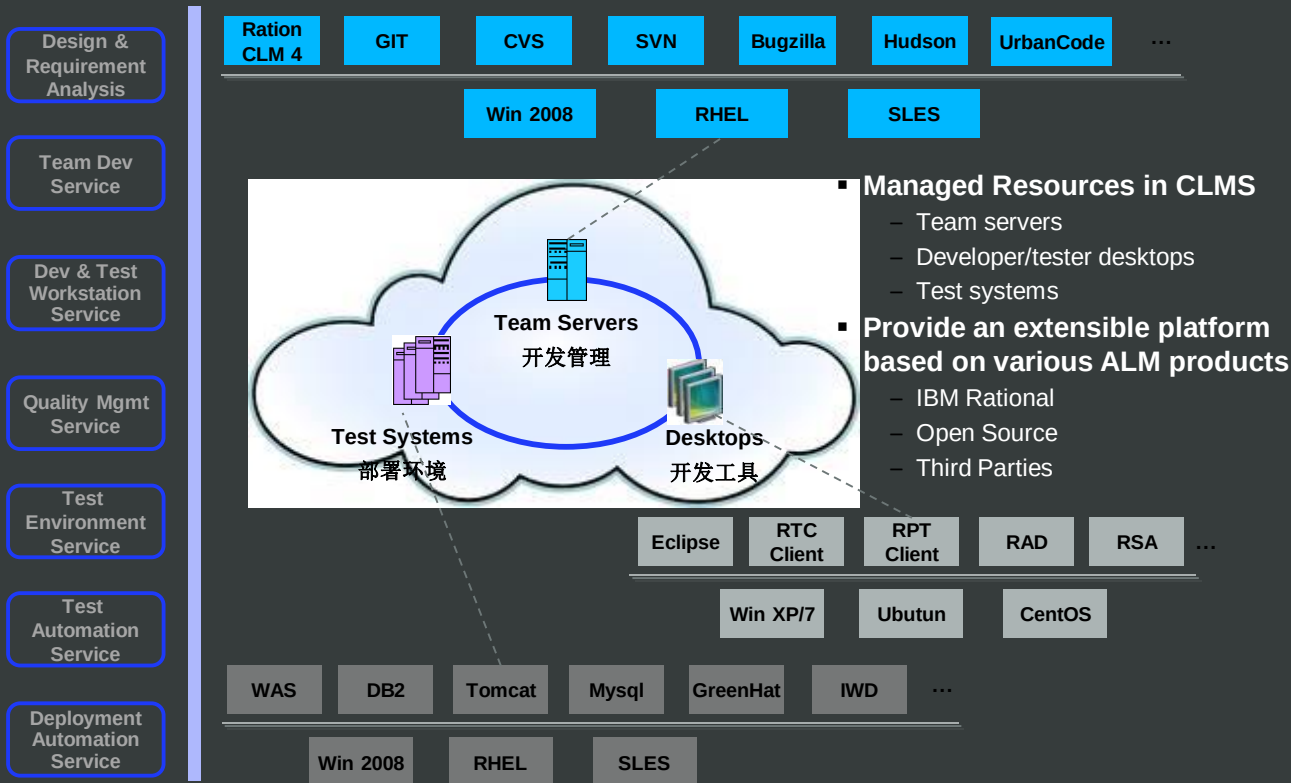




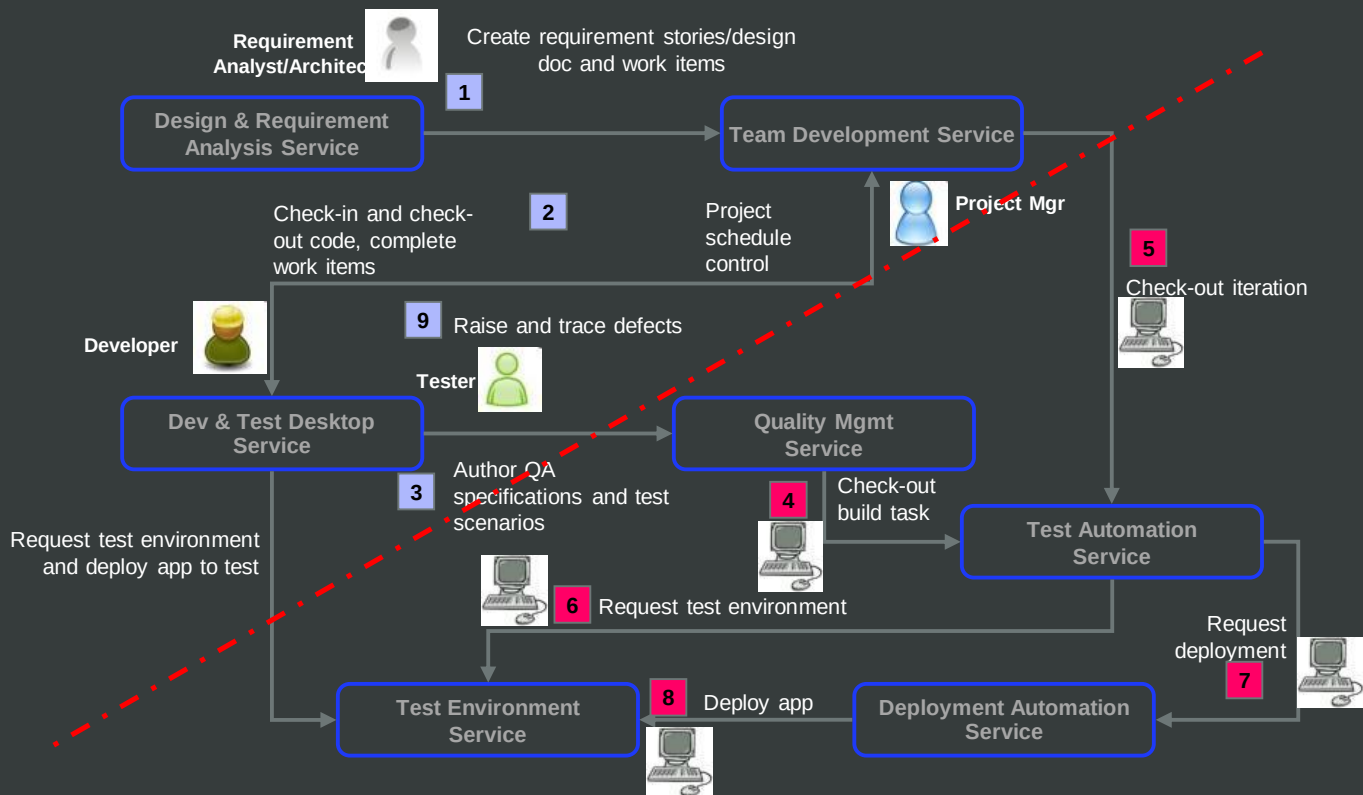




将各种工具/产品抽象成3类云资源统一管理，提供统一流程中的各种类型服务

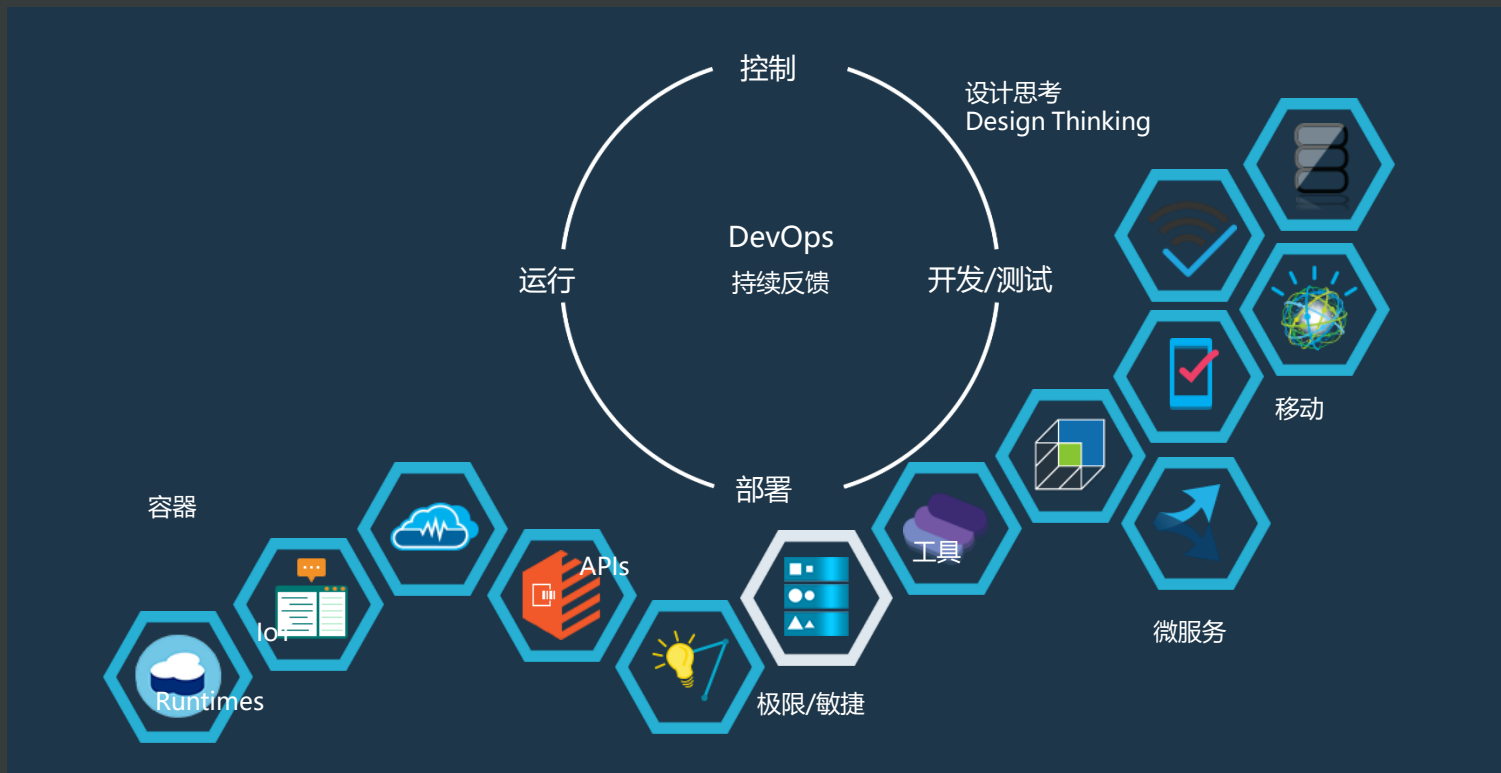


敏捷开发模式下，部分业务和流程全自动化，提高了开发/测试/预备/生产的效率



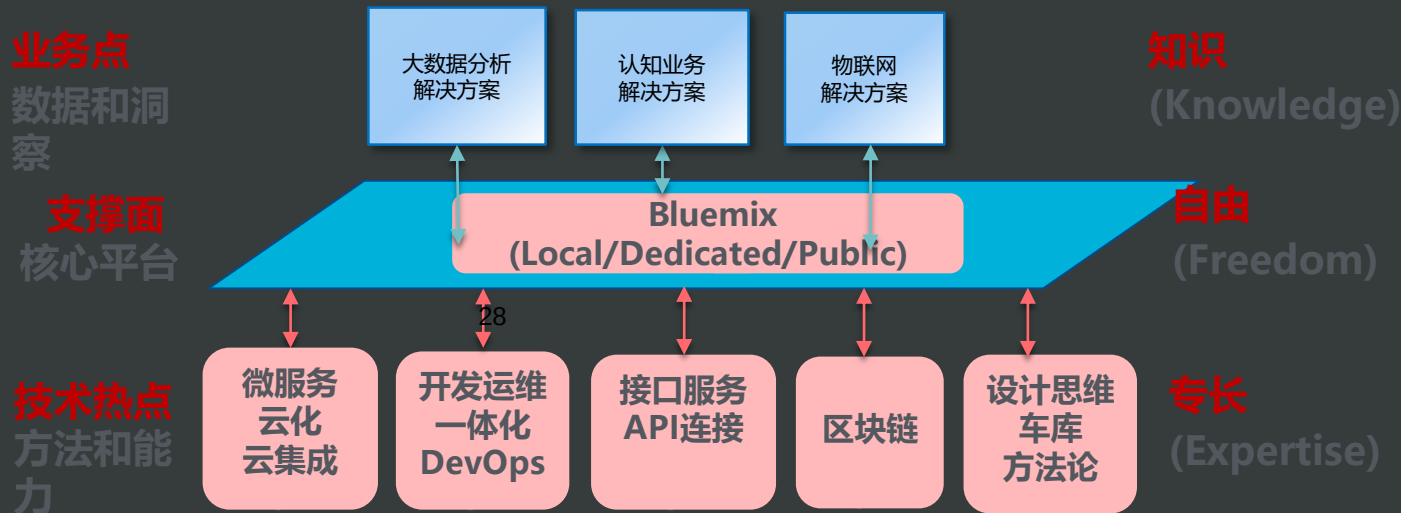
Steps 4-8 could be implemented automatically to support Continuous Development & Testing

从应用交付角度看：今天的应用交付主要有两个趋势：速度和选择的多样性

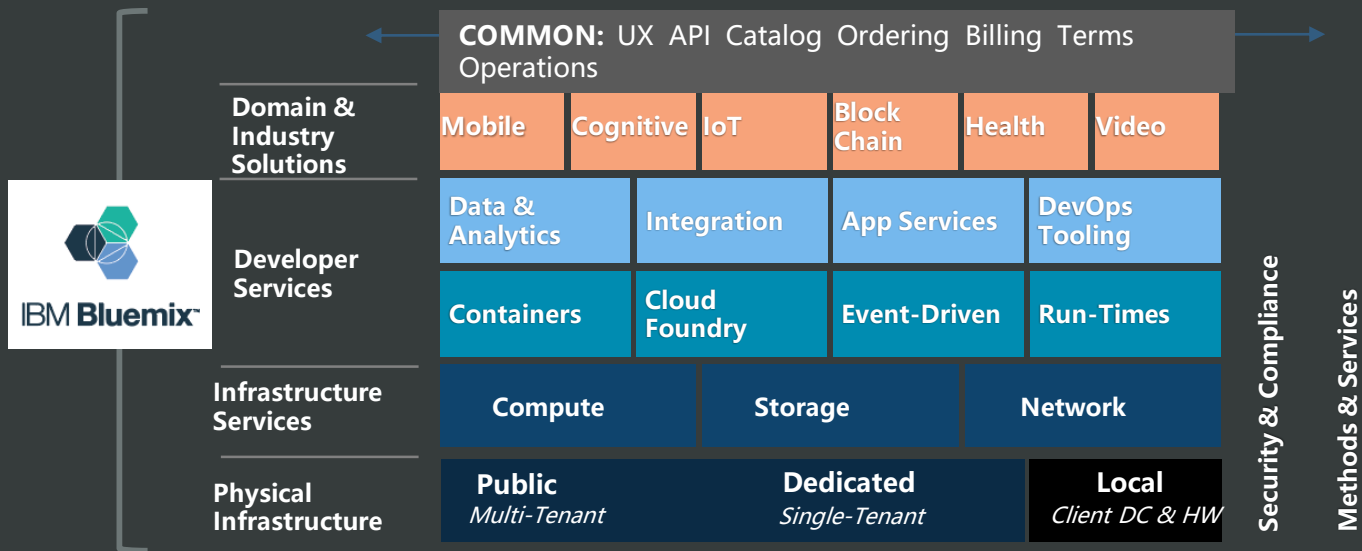


云为开发人员提供了对他们所需的APIs、服务以及运行时基础设施的快速访问途径，使他们可以将想法变成现实。

通过云平台，提升企业的能力和效率 - 面和点



- Freedom - **自由**: 这是你的核心平台、你的云，你可以自由地拥有它，放在哪里、如何跑，由你决定。
- Knowledge - **知识**: 通过利用更多的数据，以及人的参与，产生意想不到的见解和洞察。
- Expertise - **专长**: 不断创新需要新的方法和专业知识。

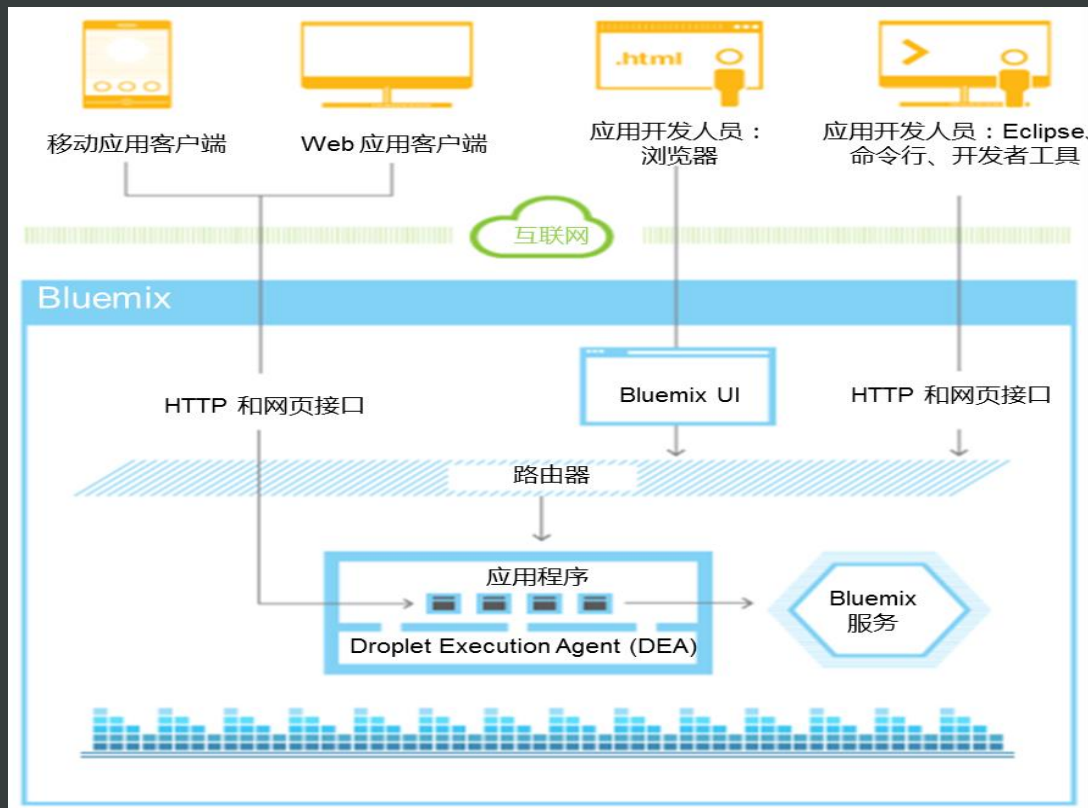


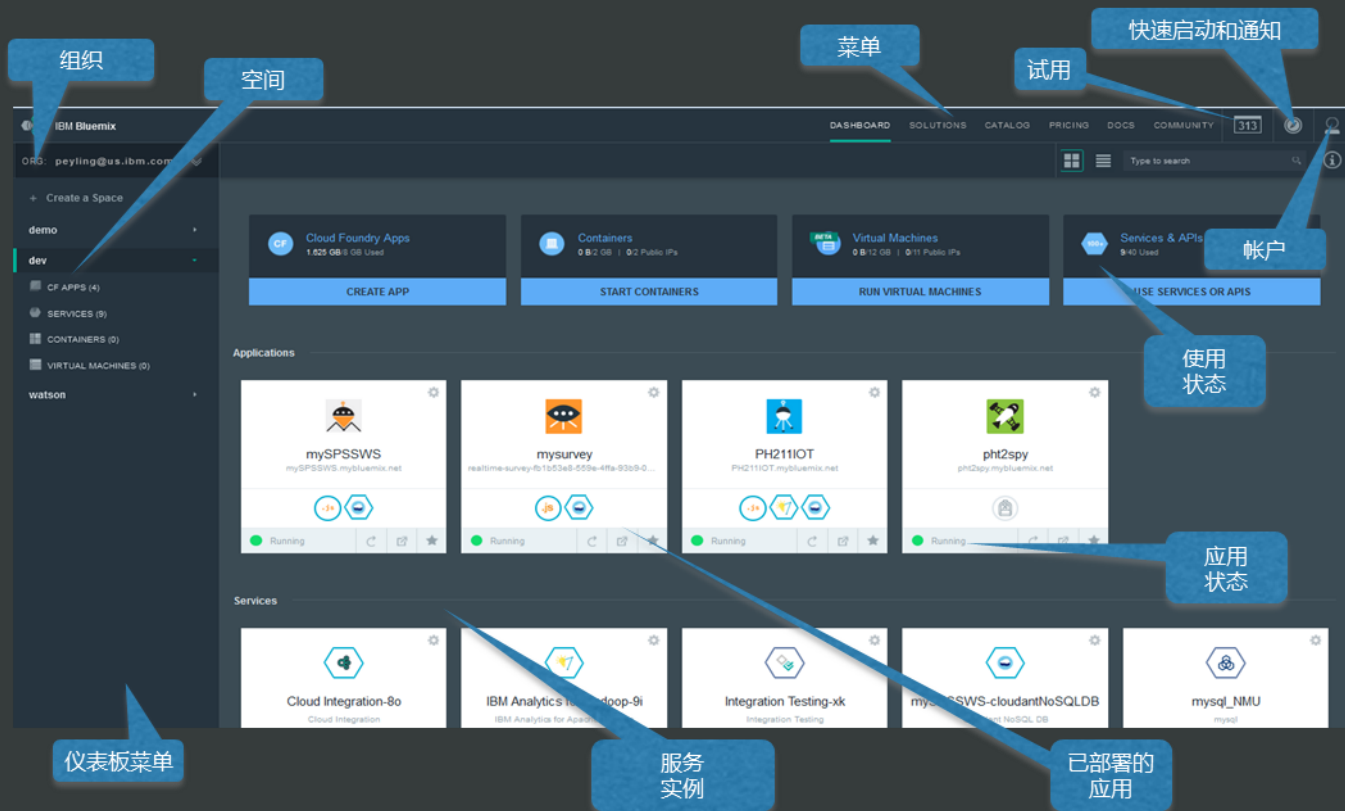
Bluemix 程序员的乐高积木式的开发测试上线全栈式平台

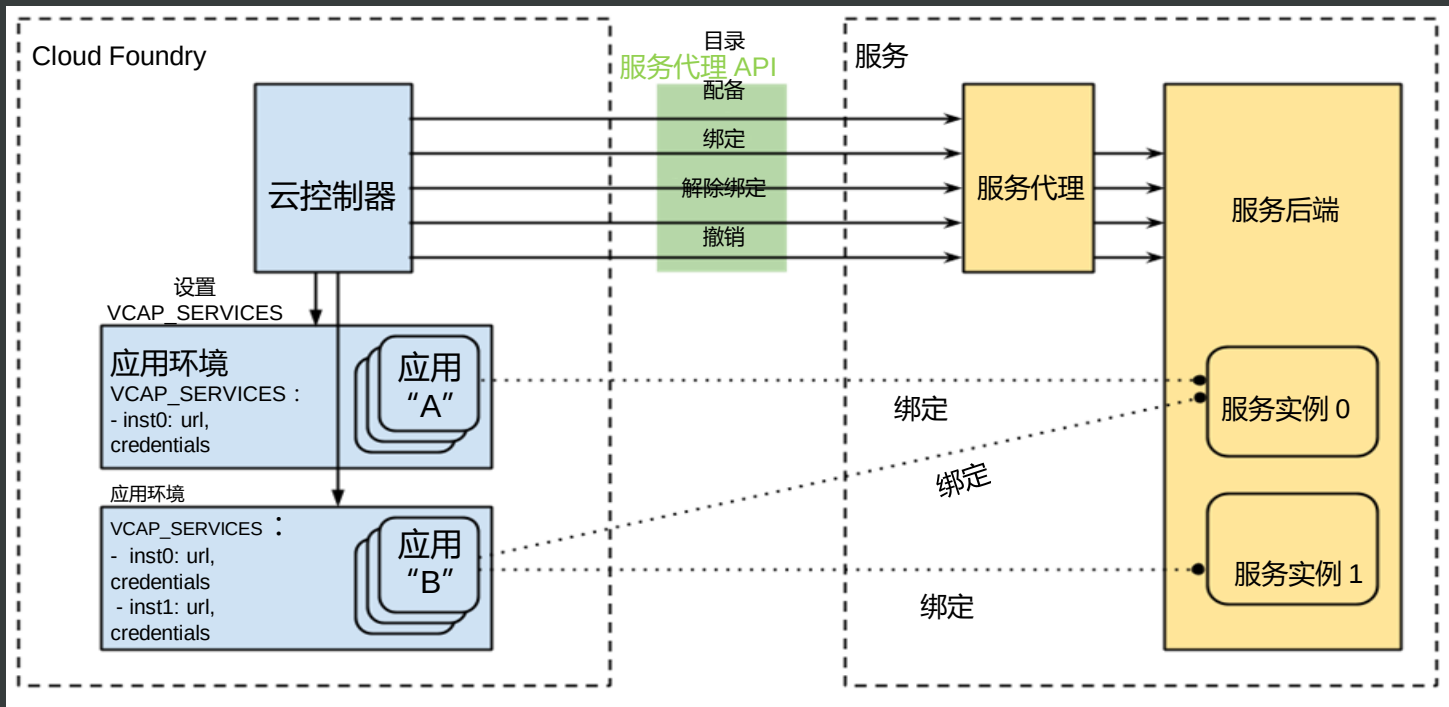


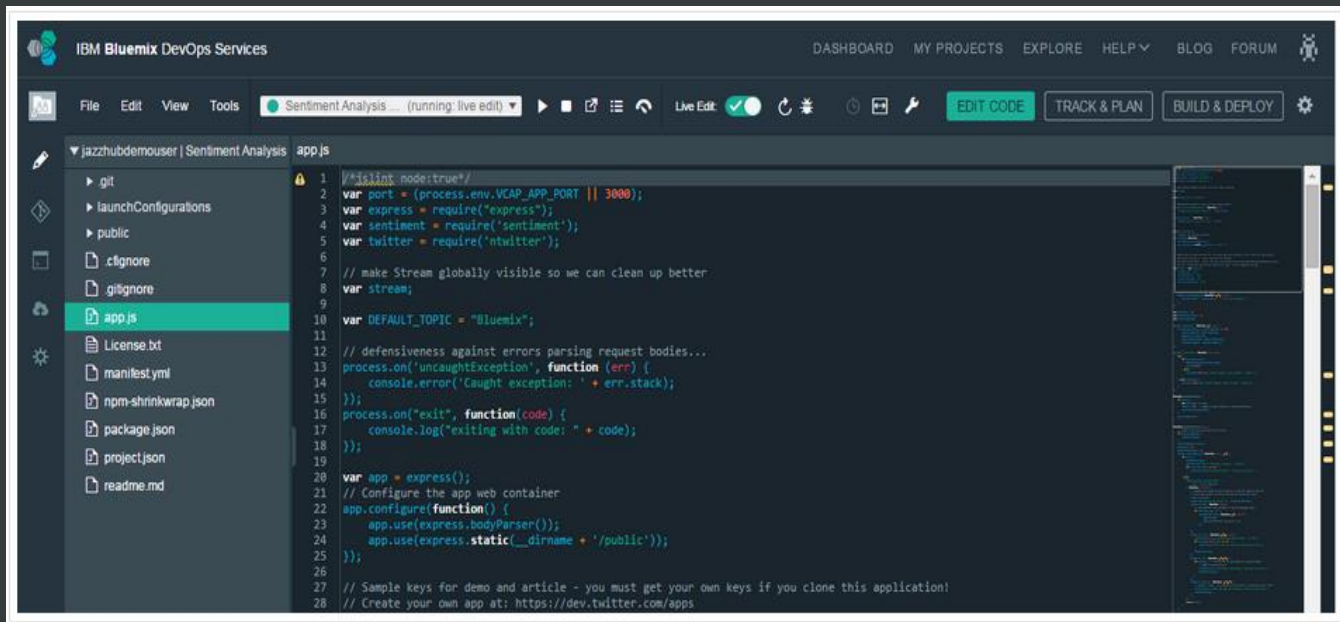
www.bluemix.net











Pipeline: All Stages

Build

INPUT
Last commit by Developer 8 minutes ago
1234: fix UI bug

JOBS
View logs and history

JOBS COMPLETED SUCCESSFULLY

- Build Succeeded 6 minutes ago
- Unit Test Succeeded 4 minutes ago

LAST EXECUTION RESULT

Build 75

Test

INPUT
Stage: Build / Job: Build
Last Input: Build 75

JOBS
View logs and history

JOBS RUNNING...

- Deploy Succeeded just now
- UI Tests Running... 15%

LAST EXECUTION RESULT

MyApp
myapp-test.mybluemix.net
View runtime log

Build 74

Production

INPUT
Stage: Build / Job: Build
Last Input: Build 68

JOBS
View logs and history

JOBS COMPLETED SUCCESSFULLY

- Deploy Succeeded 8 hours ago
- UI Tests Succeeded 8 hours ago

LAST EXECUTION RESULT

MyApp
myapp-prod.mybluemix.net
View runtime log

Build 68

Pipeline: All Stages

OSC 源创会

Opensource Innovation Meetup

IT大咖说

IT大咖说

Build Stage

STAGE NOT RUN

LAST INPUT
Not yet run

JOBS
View logs and history

- Build Not yet run

LAST EXECUTION RESULT
No results

Stage Configuration

Deploy Stage

INPUT JOBS ENVIRONMENT PROPERTIES

Input Settings

Input Type

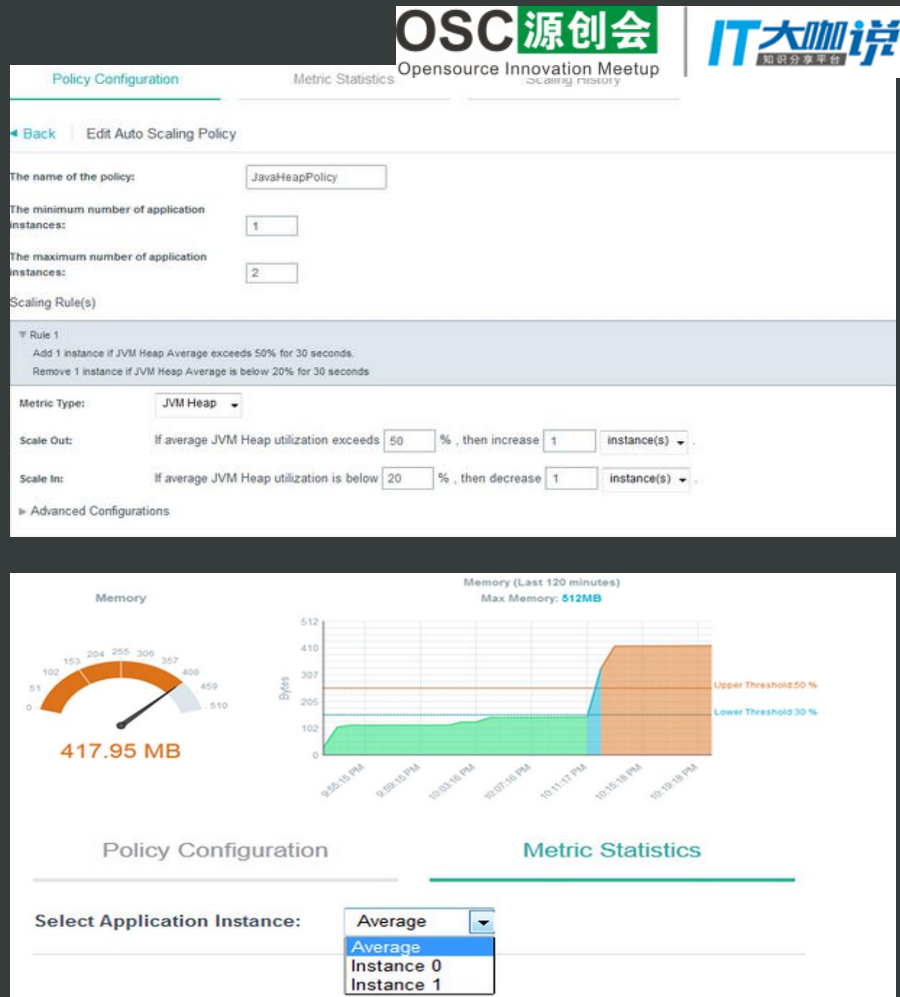
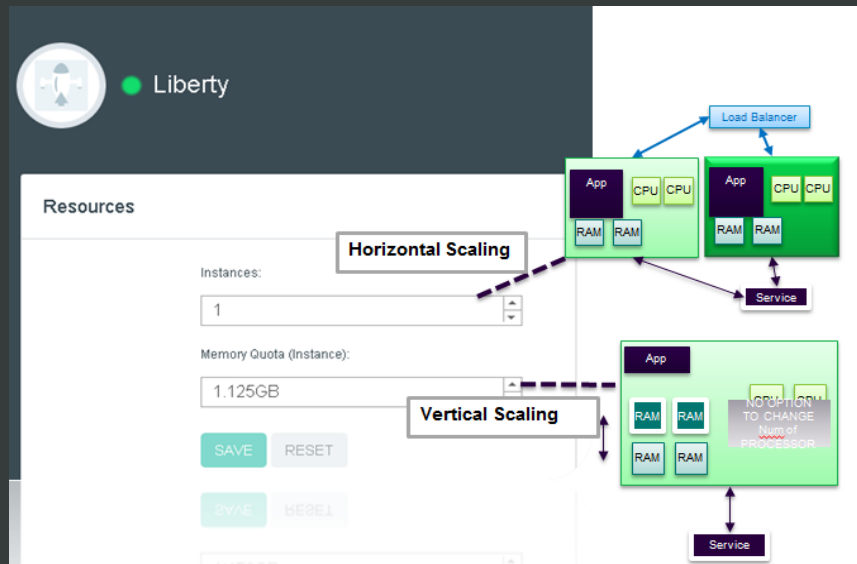
INPUT JOBS ENVIRONMENT PROPERTIES

Build

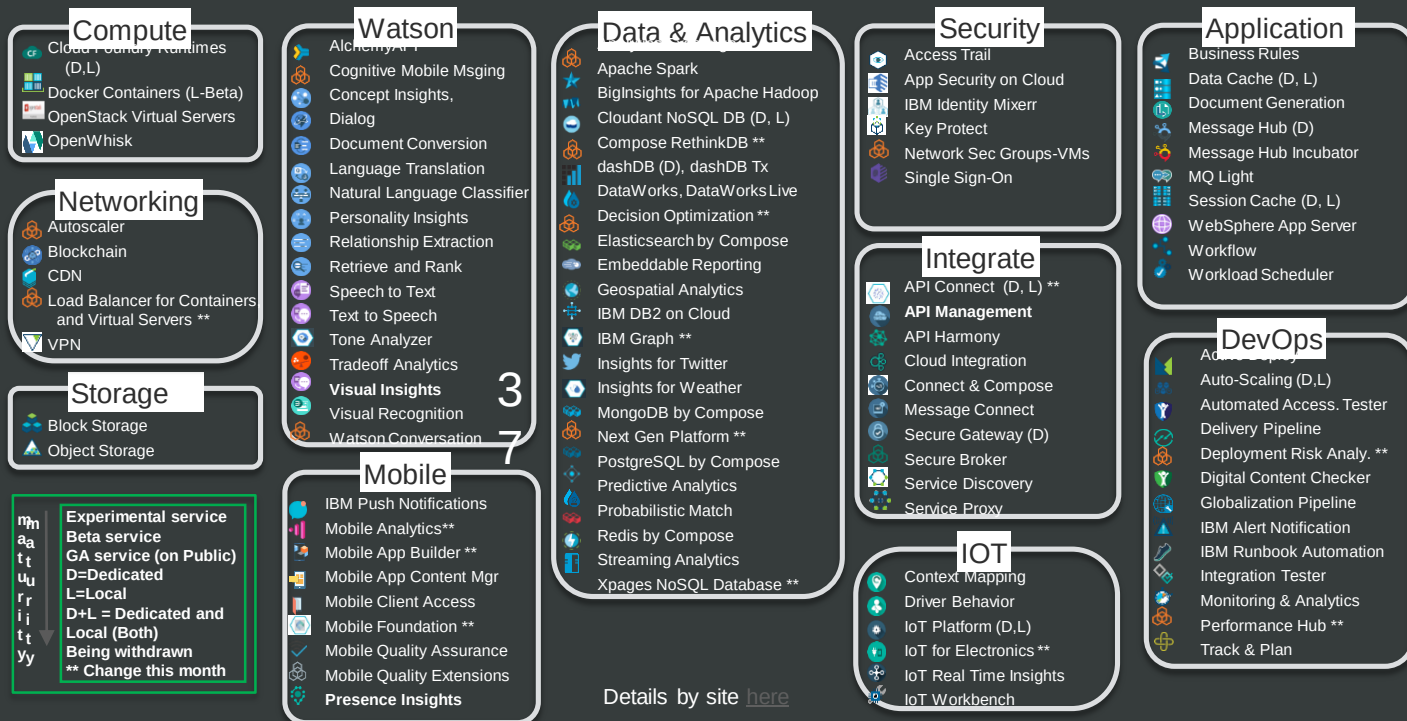
Build

SELECT JOB TYPE

- Build
- Deploy
- Test

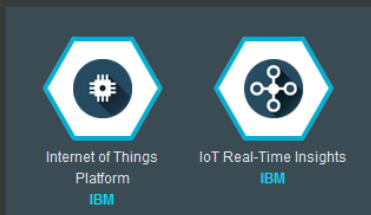


Bluemix Service Catalog 服务组件

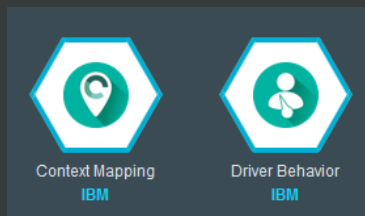


Bluemix Business Case

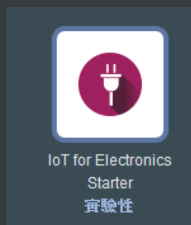
Bluemix Service 介绍说明 : IOT 物联网 : 作为物联网的接入点和平台



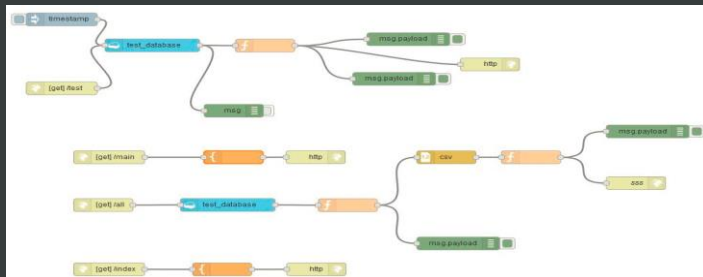
IoT Platform 物联网平台



IoT for Automotive 车联网



IoT for Electronics 智能家电



* Node.Red
物联网的大脑

* 博物馆、仓库监控



* 电梯监控

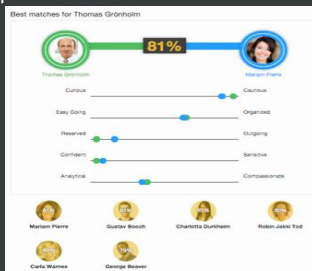
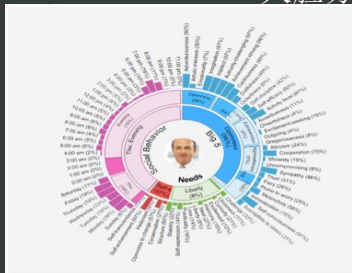


* 电能监控

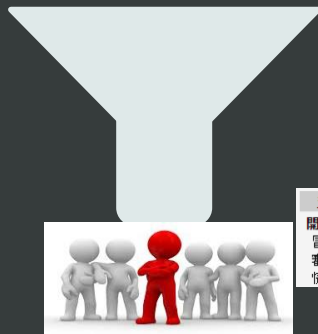


Bluemix Service 介绍说明：AI 认知运算和机器人

* 人脸分析

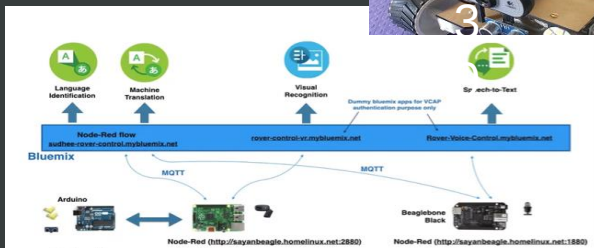


* Personal Insight 个人见解分析



五大人格特質 (Big 5)	
開放性	96%
冒險	90%
審美	62%
情緒豐富	19%

* 语音及图像识别



* Tone Analytics 语调分析

* Tradeoff Analytics 权衡分析



Bluemix Business Case

Bluemix Service 介绍说明：BlockChain 区块链 端到端的解决方案，重塑业务的交易处理方式

共享账本

智能合约

隐私保护

共识机制

为业务带来价值的区块链



社区+开发

Linux Foundation
Hyperledger Project

开放源代码

Permission | Privacy
Confidential |
Auditable

开源管理



云计算

Blockchain on
Bluemix

在 SoftLayer 或大型主机
System z 上的区块链增值
管理服务

Identity | Consensus |
Audit | System
Integration |
Hardware-assist for
Performance &
Security



企业

Blockchain Solutions
Bluemix Garage

金融服务、医疗、供应链与物
联网的区块链服务

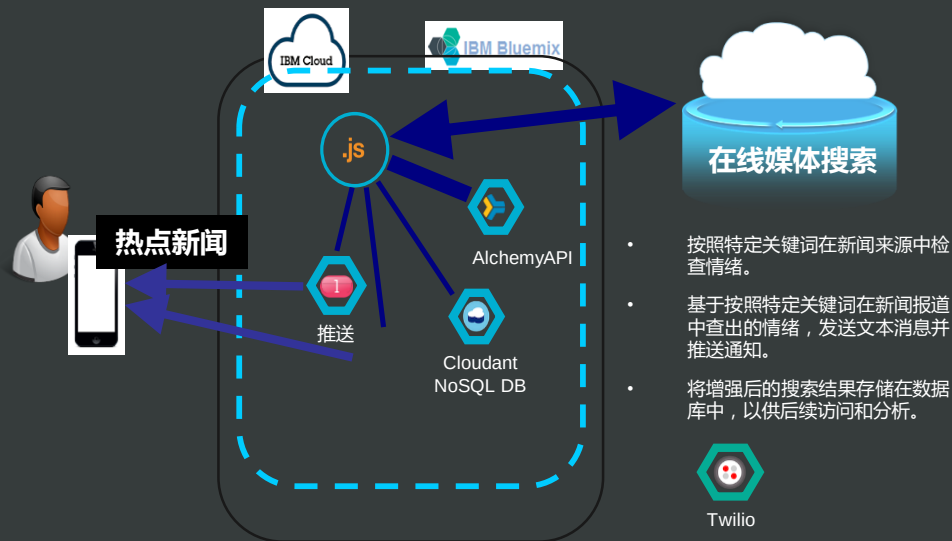
Finance | Capital Markets

Bluemix Garage
纽约 | 伦敦 | 新加坡 | 东京

Blockchain 最佳实践

Bluemix Dev Case

使用 AlchemyAPI



AlchemyLanguage

OSC 源创会

IT大咖说

实时文本分析，以进行自OpenSource Innovation Meetup

12 个 API

- 实体提取
- 情绪分析
- 概念标记
- 关键词提取
- 关系提取
- 分类
- 作者提取
- 语言检测
- 文本提取
- 微格式解析
- 来源检测
- 链接数据支持

5 个 API

- 图像链接提取
- 图像标记
- 面部检测和识别
- 联合调用

关键词提取 API

- **API 调用**：URLGetRankedKeywords、HTMLGetRankedKeywords、TextGetRankedKeywords
- **输出**：用于索引数据、生成标记云或用于搜索的内容中的重要话题。
- **参数**：(apikey、text、url、outputMode、maxRetrieved keywords、keywordExtractMode、sentiment)
- **响应**：(status、language、url、relevance、text、sentiment、knowledgeGraph、status Info)

处理的文本：Since I am in New York to participate in the Data BootCamp on April 10th and Mentor hackers in NASA Space App Challenge on April 11th and 12th.I cannot be with my team to develop content for certification.

响应对象：{"status":"OK","usage":"By accessing AlchemyAPI or using information generated by AlchemyAPI, you are agreeing to be bound by the AlchemyAPI Terms of Use: http://www.alchemyapi.com/company/terms.html", "totalTransactions":"2","language":"english","keywords":[{"relevance":"0.904191","sentiment":{"type":"neutral"}, "text":"Space App Challenge"}, {"relevance":"0.736453","sentiment":{"type":"neutral"}, "text":"Mentor hackers"}, {"relevance":"0.664088","sentiment":{"type":"neutral"}, "text":"Data BootCamp"}, {"relevance":"0.459154","sentiment":{"type":"neutral"}, "text":"New York"}, {"relevance":"0.233081","sentiment":{"type":"neutral"}, "text":"NASA"}]}



购买旅游出行商品！

对应的： MVP 最小可用品 ★



作为李莉，我希望

- 提供 **旅行推荐清单**，包含出行用品和家中生活必需品；
- 在我旅行途中，及时了解**孩子和父母的生活必需品**是否得到及时补充，并提供上门服务。

车库方法论、逐步实现创新

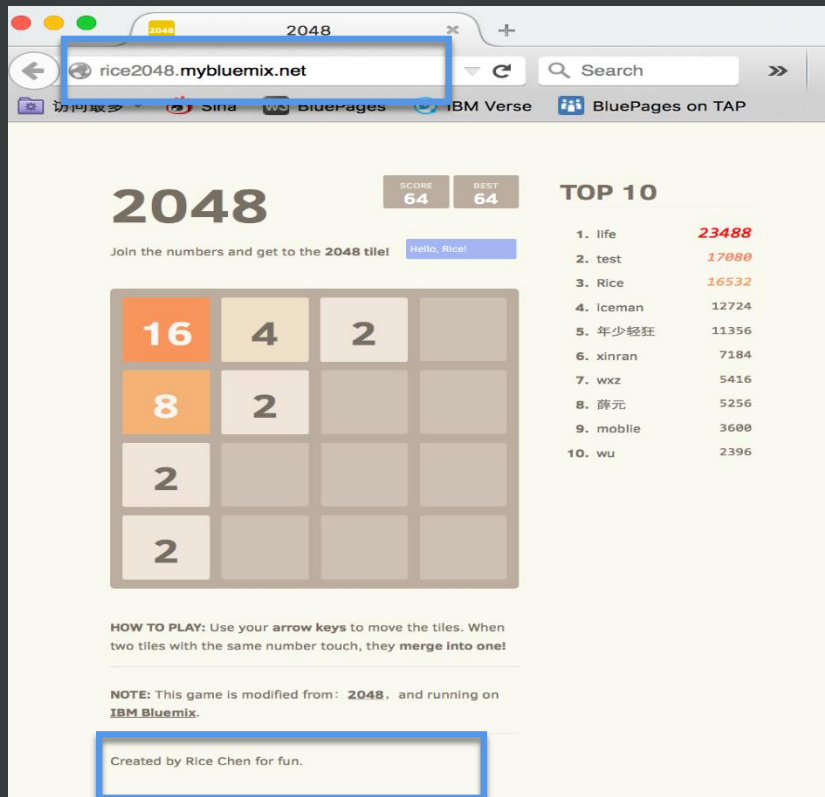
Bluemix Dev Case

快速开发一个 Bluemix 应用

开发一个应用
rice2048.mybluemix.net

请立即 开始你的Bluemix 之旅
www.bluemix.net

每周2万新注册用户
每天15亿次服务调用



Bluemix Dev Case

快速开发一个 Bluemix 应用

Steps by steps to run 2048 on Bluemix

1. Connect to Bluemix with CF command line tool.
2. Run "cf push app_name --no-start" to deploy app to Bluemix and not start app first
3. Login Bluemix UI with your Bluemix ID: <http://www.bluemix.net>
4. Click into app "app_name" and click "Add a Service" to add mysql service to this application. Click "OK" to re-stage app.
5. Start app from Bluemix UI
6. Access application with http://app_name.mybluemix.net
7. Have a try with my deployment: <http://2048-bluemix.mybluemix.net/>

```
1480 ;(function($){
1481   $.fn.serializeArray = function() {
1482     var result = [], el
1483     $([].slice.call(this.get(0).elements)).each(function(){
1484       el = $(this)
1485       var type = el.attr('type')
1486       if (this.nodeName.toLowerCase() != 'fieldset' &&
1487         !this.disabled && type != 'submit' && type != 'reset' && type != 'button' &&
1488         ((type != 'radio' && type != 'checkbox') || this.checked))
1489         result.push({
1490           name: el.attr('name'),
1491           value: el.val()
1492         })
1493     })
1494     return result
1495   }
1496
1497   $.fn.serialize = function(){
1498     var result = []
1499     this.serializeArray().forEach(function(elm){
1500       result.push(encodeURIComponent(elm.name) + '=' + encodeURIComponent(elm.value))
1501     })
1502     return result.join('&')
1503   }
1504
1505   $.fn.submit = function(callback) {
1506     if (callback) this.bind('submit', callback)
1507     else if (this.length) {
1508       var event = $.Event('submit')
1509       this.eq(0).trigger(event)
1510       if (!event.isDefaultPrevented()) this.get(0).submit()
1511     }
1512     return this
1513   }
1514
1515 })(Zepto)
```

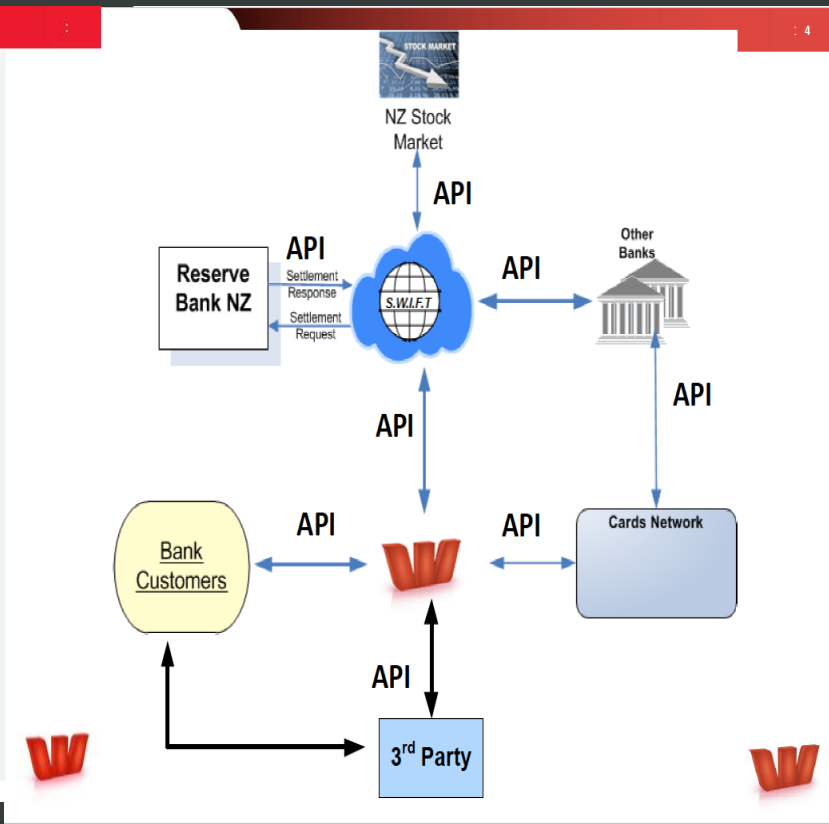
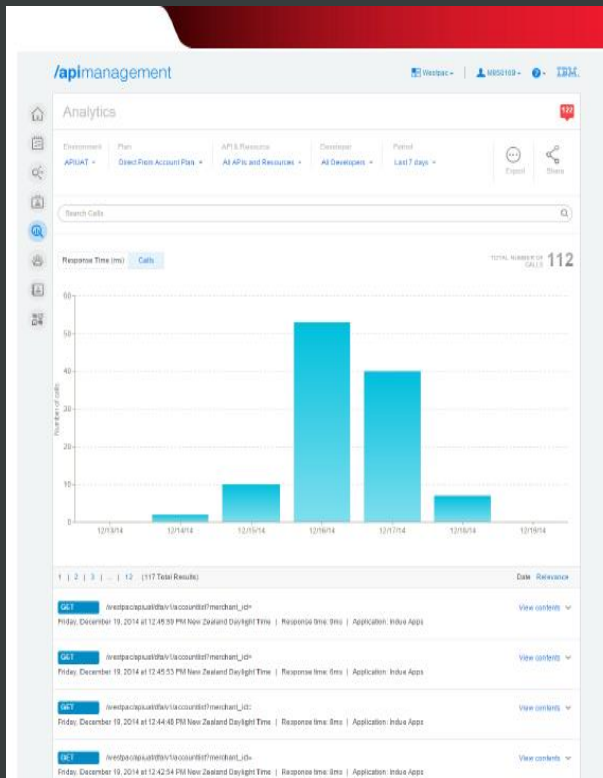
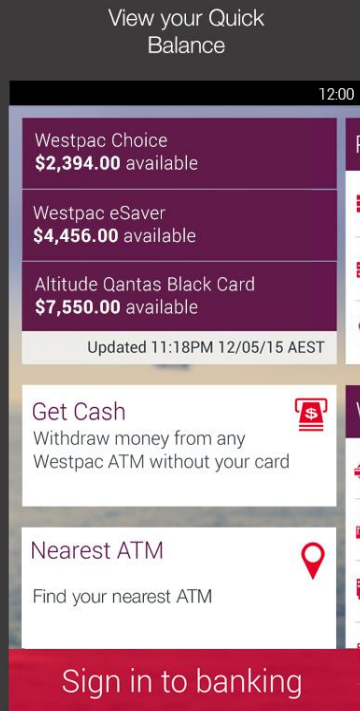
use ClearDB MySQL DB for database

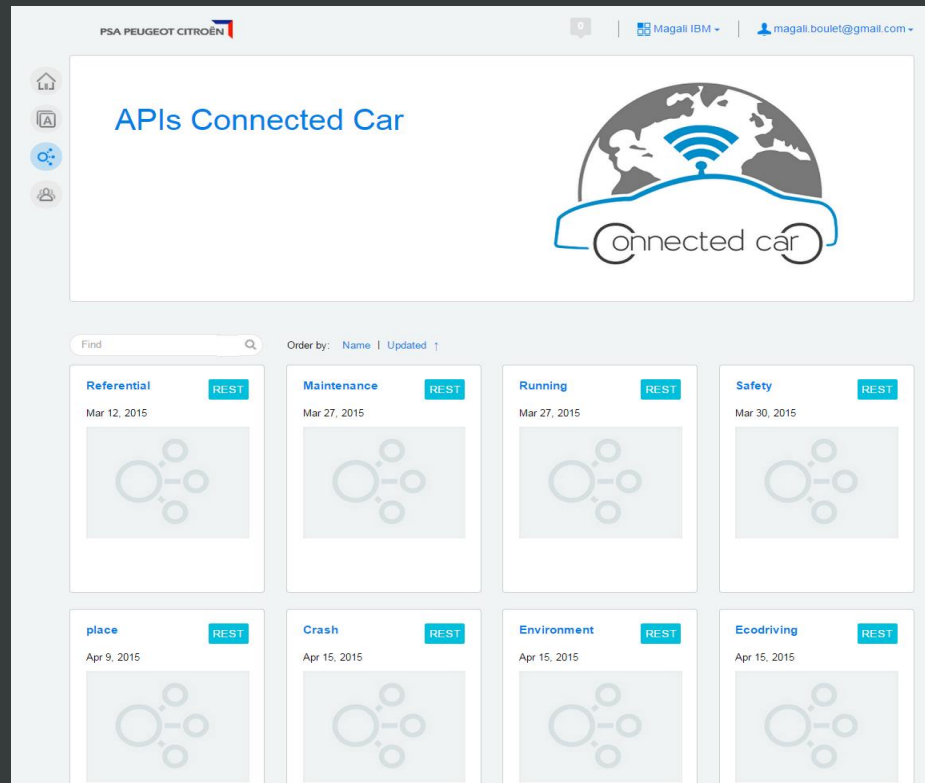
澳洲某银行的API能力平台

业务洞察

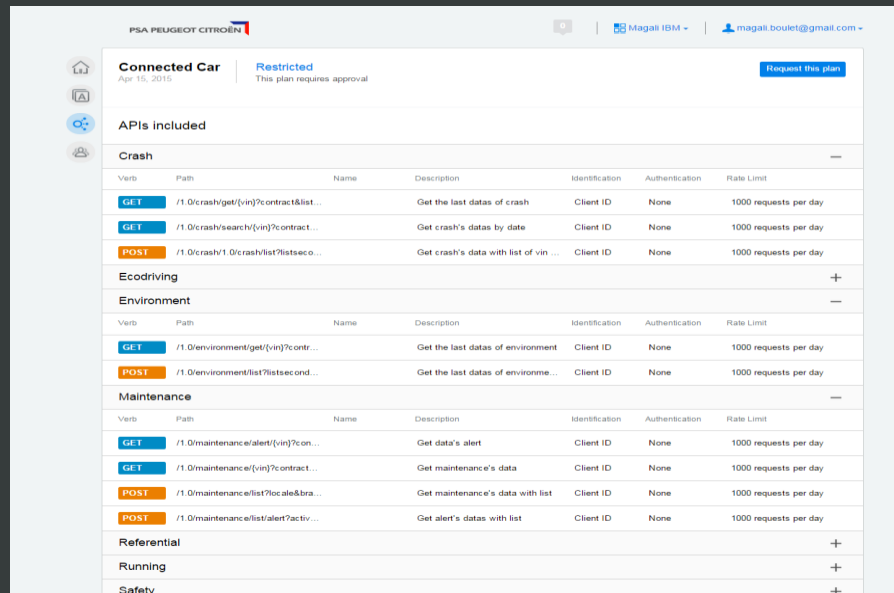
在线安全电子支付

移动银行





- ✓ 车载设备和移动应用的“随时在线”的低延迟交互
- ✓ 车载设备数据通过API发布在安全的开发者门户
- ✓ 内外部开发人员可以利用车载数据开发各种应用，特别是移动应用
- ✓ 激发各种创新的商业模式



某旅游集团的创新平台，构建区域智慧旅游服务体系



Thank You !

蒋旭春