

16:30 - 17:00 高效 pipeline 助力传统软件向 cloud 迁移



樊懋程：SAP中国研究院技术专家。先后负责SAP Business One，Success Factors产品的自动化和性能测试框架设计开发工作，近期主要专注CI/CD过程效率提升和软件质量把控。

高效PIPELINE

助力传统软件向cloud迁移

樊懋程
SAP中国研究院

面临的问题：

- ❑ Monolithic
- ❑ 多租户支持/隔离
- ❑ 模块/代码复杂，API不清晰，不友好
- ❑ Mobile
- ❑ 难以扩展
- ❑ 性能
- ❑ 部署困难，需要自动化运维 (Failover)

解决思路：



新问题：

- ❑ 3 service -> ~20 service >~70 service
- ❑ API兼容性问题
- ❑ 集成失败，Regression，导致无可测版本
- ❑ 质量黑盒
- ❑ 发布周期过长
- ❑ 多个云平台需要部署
- ❑ service配置管理

解决思路：

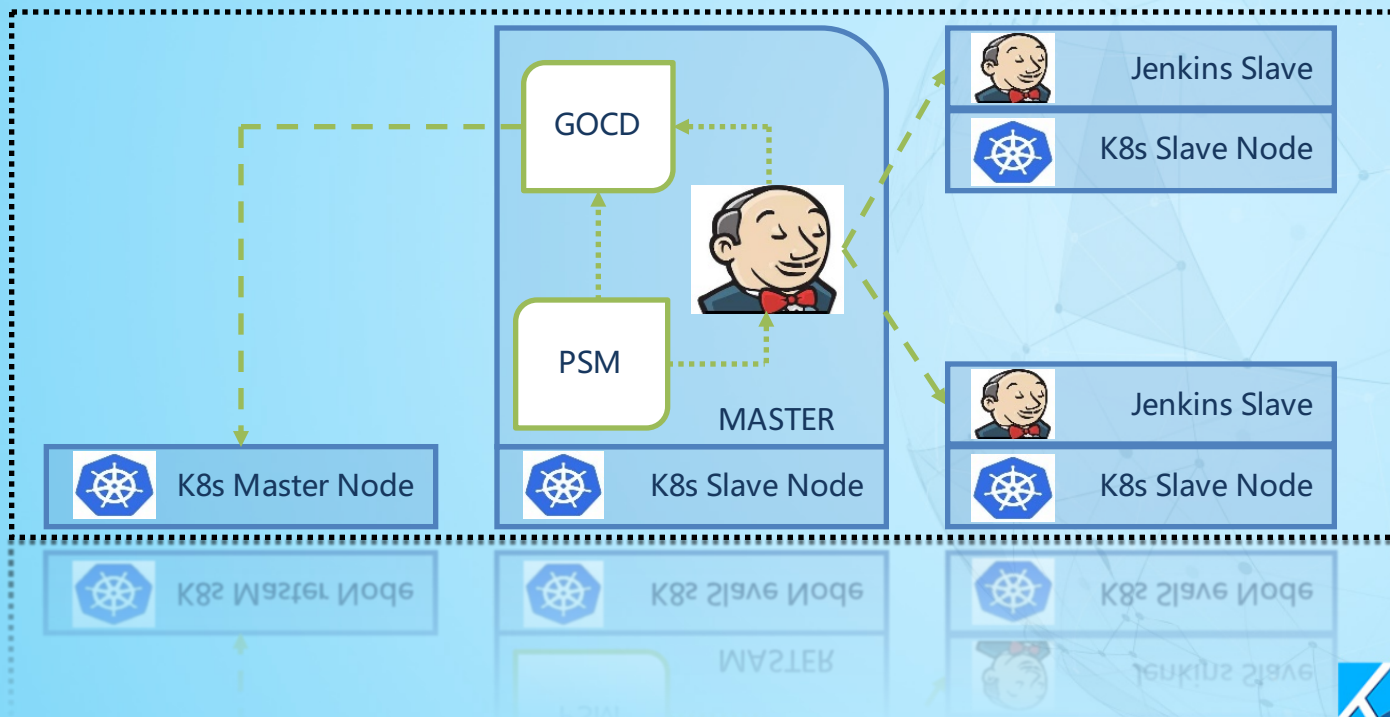
高效 CI/CD PIPELINE



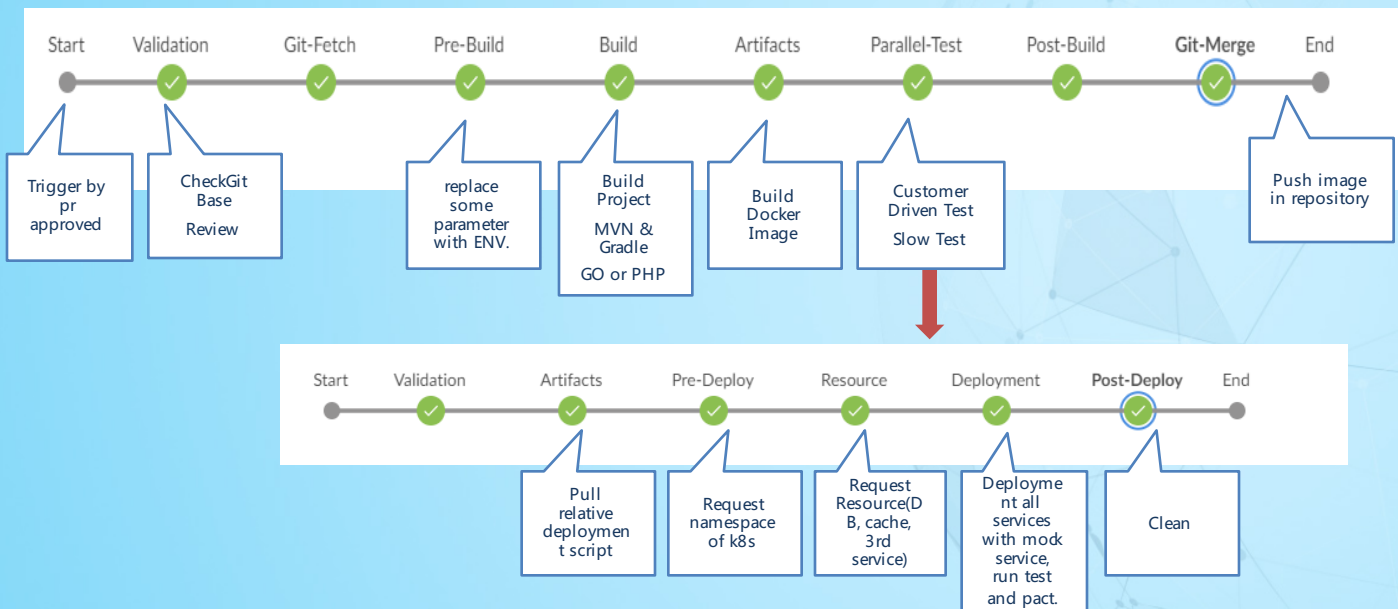
定义 -- 高效PIPELINE

- ❑ 全程自动化，端到端，打通多个云平台（AWS，AliYun，SAP Internal...）
- ❑ Everything In Pipeline
- ❑ 自定义PIPELINE流程配置
- ❑ FAST – 快速实施，快速测试，快速集成，快速反馈，快速发布
- ❑ 全流程监控，收集反馈

PIPELINE 架构

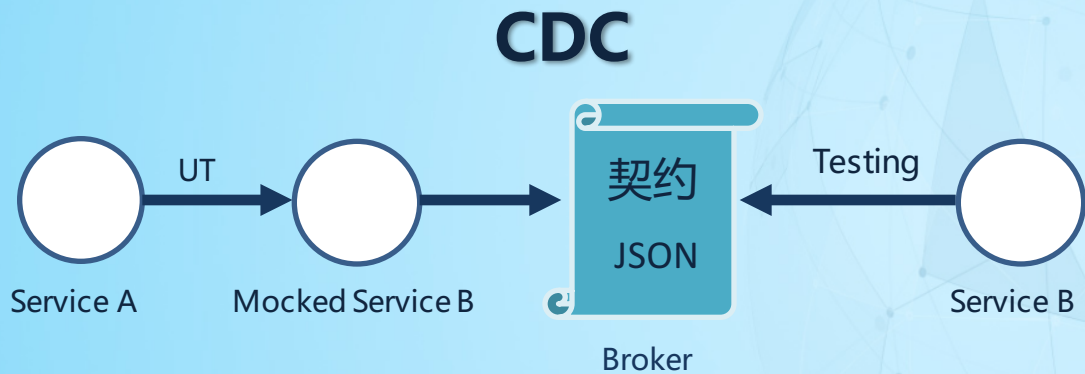


CI 流程

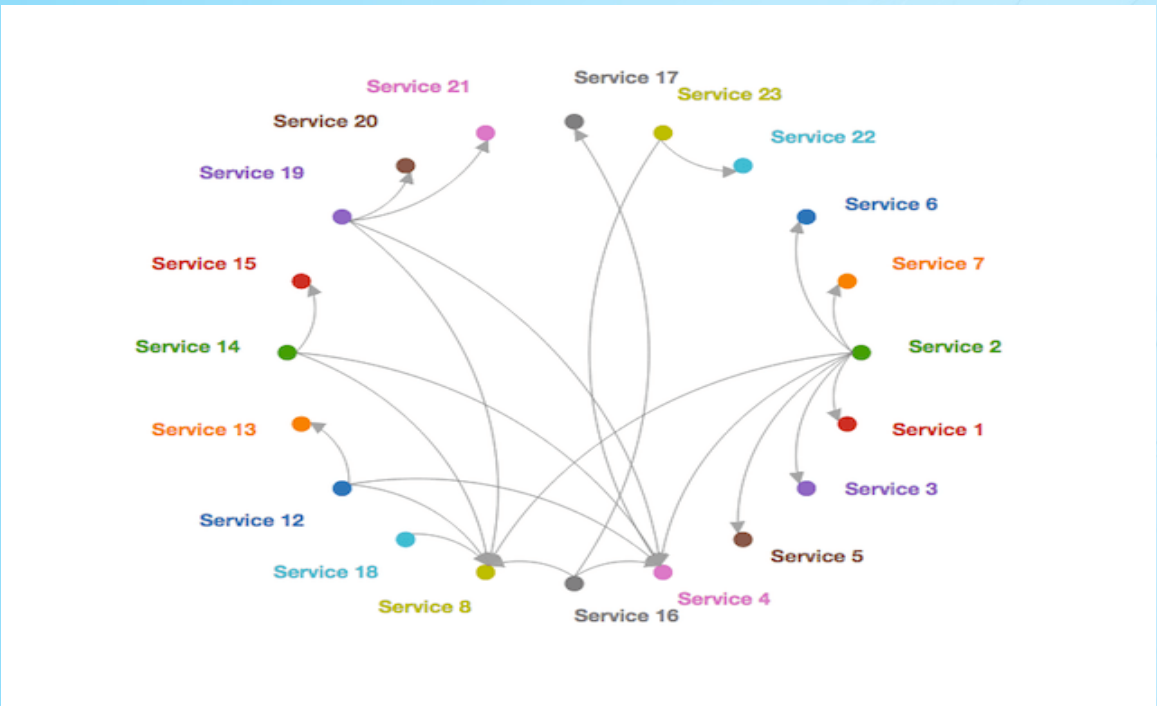


CDC

Consumer-Driven Contracts



CDC -- PACT



CDC -- PACT

```
@Pact(consumer = "messaging-app")
public RequestResponsePact pactUserExists(PactDslWithProvider builder) {
    return builder.given("User 1 exists")
        .uponReceiving("A request to /users/1")
            .path("/users/1")
            .method("GET")
            .willRespondWith()
                .status(200)
                .body(LambdaDsl.newJsonBody((o) -> o
                    .stringType("name", "user name for CDC")
                )
            )
        .build()
    .toPact();
}

@PactVerification(fragment = "pactUserExists")
@Test
public void userExists() {
    final User user = userServiceClient.getUser("1");
    assertThat(user.getName(), isEqualTo("user name for CDC"));
}
}
```

```
}
    .stringType("name", "user name for CDC");
    final User user = userServiceClient.getUser("1");
    public void userExists() {
    @Test
    @PactVerification(fragment = "pactUserExists")
```

CDC -- PACT



contract.json

CDC -- PACT

```
@Service
public class UserService {

    public User findUser(String userId) {
        return User.builder()
            .id(userId)
            .legacyId(UUID.randomUUID().toString())
            .name("MossinFan")
            .role(UserRole.ADMIN)
            .lastLogin(new Date())
            .friend(Friend.builder().id("2").name("Li Qian").build())
            .build();
    }
}
```

CDC -- PACT

```
Run: ContractTest x
Tests passed: 1 of 1 test - 1 s 194 ms
ContractTest (de.kreuzwerker.cdc.userser 1 s 194 ms)
  messaging-app - A request to /users/1 1 s 194 ms
    Verifying a pact between messaging-app and user-service
    Given User 1 exists
    A request to /users/1
    returns a response which
    has status code 200 (OK)
    includes headers
    "Content-Type" with value "application/json; charset=UTF-8" (OK)
    has a matching body (OK)
```


CI 配置

```
Project - idl-cache ~/git_repo
├── .idea
├── deploy
├── idl-cache-api
├── idl-cache-build
├── idl-cache-service
├── idl-cache-ui
├── test
├── .gitignore
├── build.sh
├── build.yaml
├── CONTRIBUTING.md
├── Dockerfile
├── Jenkinsfile
├── rebel.xml
├── README.md
├── External Libraries
└── Scratches and Consoles

1  name: idl-cache
2
3  tasks:
4  - name: prebuild
5    command:
6      - sed -i 's/{{GIT_COMMIT}}/{{GIT_COMMIT}}/g' deploy/*.yaml
7      - sed -i 's/{{GIT_BRANCH}}/{{GIT_BRANCH}}/g' deploy/*.yaml
8  - name: build
9    command:
10     - ./build.sh
11     timeout: 20m
12  - name: artifacts
13    kind: Docker
14    images:
15     - name: bizx
16       path: .
17     - name: idl-cache-deploy
18       path: deploy
19     - name: init-cache-config
20       path: test/init-cache-config
21     timeout: 30m
22  - name: slowtest
23    kind: Job
24    timeout: 45m
```

```
├── deploy
├── test
│   ├── aep-test-job.yaml
│   ├── bizx1-deploy.yaml
│   ├── bizx1-service.yaml
│   ├── bizx2-deploy.yaml
│   ├── bizx2-service.yaml
│   ├── bizx-configmap.yaml
│   ├── cache-config-configmap.yaml
│   ├── cache-config-deployment.yaml
│   ├── cache-config-ingress.yaml
│   ├── cache-config-svc.yaml
│   ├── deploy.yaml
│   ├── Dockerfile
│   ├── init-cache-config.yaml
│   ├── init-cache-config-job.yaml
│   ├── node-manager-configmap.yaml
│   ├── node-manager-deployment.yaml
│   └── node-manager-svc.yaml
```

CI 增强版配置

```
Project - au-cdp ~/git_repo/au-cdp
└─ .idea
└─ au-cdp-api
└─ au-cdp-build
└─ au-cdp-service
└─ au-cdp-ui
└─ au-cdp-web
└─ .dockerignore
└─ .gitignore
└─ .pipeline.yaml
└─ component.jenkinsfile
└─ component.properties
└─ CONTRIBUTING.md
└─ Dockerfile
└─ dynamicKarma.js
└─ Jenkinsfile
└─ rebel.xml
└─ README.md
└─ reportCDPDependency.dt
└─ External Libraries
└─ Scratches and Consoles

1 | |---
2 | | apiVersion: v1beta
3 | | kind: bizx-component
4 | | metadata:
5 | |   name: au-cdp
6 | | spec:
7 | |   test-env:
8 | |     - name: hana
9 | |     - name: bizx
10 | |     - name: bizx-db-upgrade
11 | |     - name: zookeeper
12 | |     - name: kafka
13 | |   tests:
14 | |     - name: qray
15 | |       type: qray
16 | |       timeout: 40m
17 | |       values:
18 | |         filter: has("cdp_cdpipeline")
19 | |         props: companyId=hhtCDPV1207,adminExternalId=admin,adminId=admin
20 | |   test-steps:
21 | |     - [qray]
```

CD PIPELINE

- ❑ 不同部署环境的权限控制
- ❑ Cloud Platform Provider
- ❑ 配置文件管理

CD PIPELINE

- ❑ Dev-Info System
- ❑ Spinnaker
- ❑ *Configuration as a Service (Vault)

Dev-Info System

Dev Info Administration

Site administration

AUTHENTICATION AND AUTHORIZATION

Groups + Add Change

CONFIGURATIONS

Branches + Add Change

Builds + Add Change

Developers Change

Environments + Add Change

Locks + Add Change

Preferences + Add Change

Repositories + Add Change

DJANGO_CELERY_RESULTS

Task results + Add Change

OPENID CONNECT PROVIDER

Authorization Codes Change

Clients + Add Change

RSA Keys + Add Change

Tokens Change

Groups Change

Users + Add Change

Groups + Add Change

Authorization Codes Change

OPENID CONNECT PROVIDER

Groups + Add Change

Dev-Info System

Home > Configurations > Environments

Select environment to change

Action: 0 of 4 selected

<input type="checkbox"/>	NAME	CLUSTER	DEVELOPERS	UNLOCKED SINCE	IS ACTIVE
<input type="checkbox"/>	dm	Testing k8s	[REDACTED]	June 14, 2018, 5:14 a.m.	✓
<input type="checkbox"/>	jenkins-*	Local CoreOS	[REDACTED]	June 14, 2018, 11:15 a.m.	✓
<input type="checkbox"/>	sf	dev	Mossin Fan [068528]	Nov. 29, 2018, 9:35 a.m.	✓
<input type="checkbox"/>	smec	Coverage Cloud	[REDACTED]	June 14, 2018, 7:26 a.m.	✓

Deploy-Job

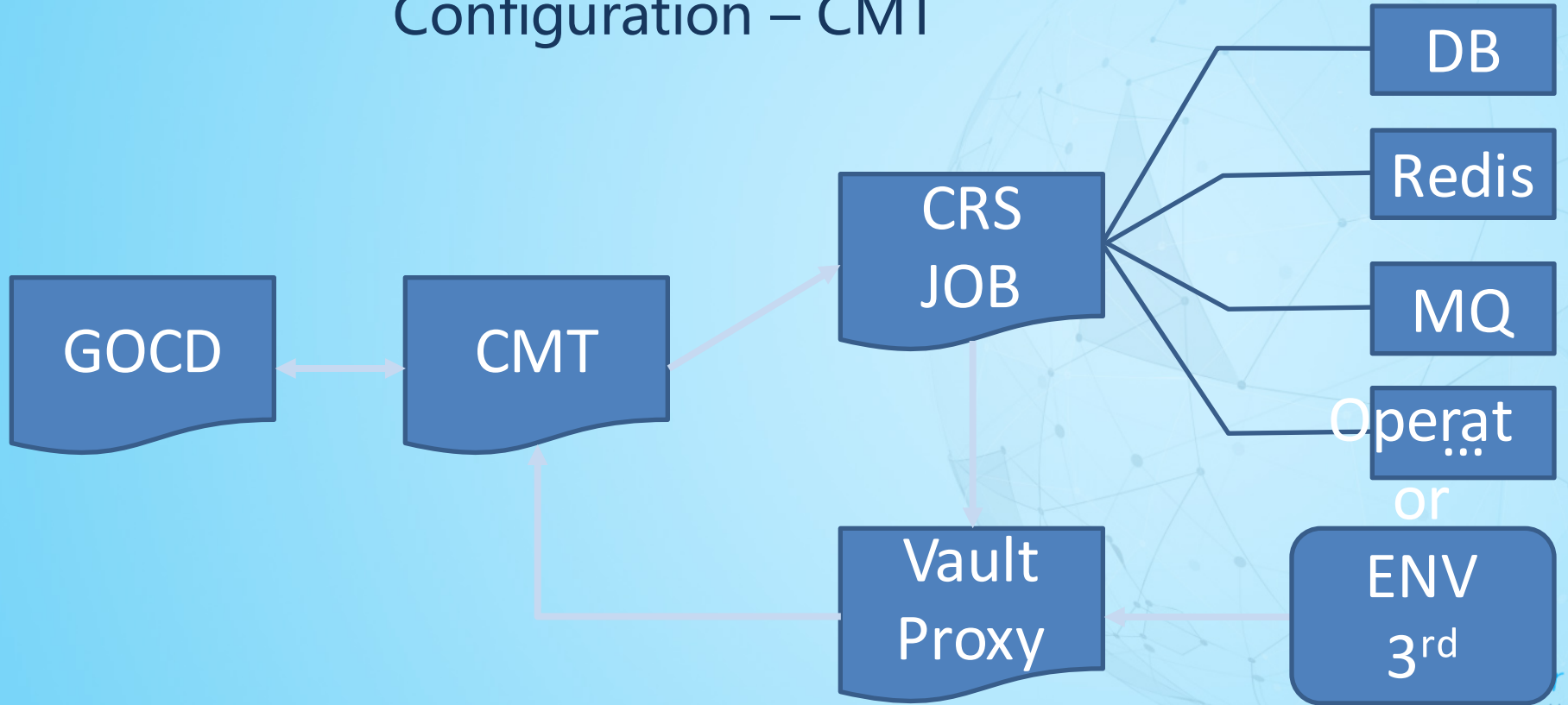
Parameters	
NAMESPACE	smec.minikube
ACTION	upgrade
REPO	anywhere/cmt/tree/master
ENV	{"VAULT_SERVER":"1","NEED_BACKUP":"1","STORAGE_CLASS":"standard"}

Pipeline deploy

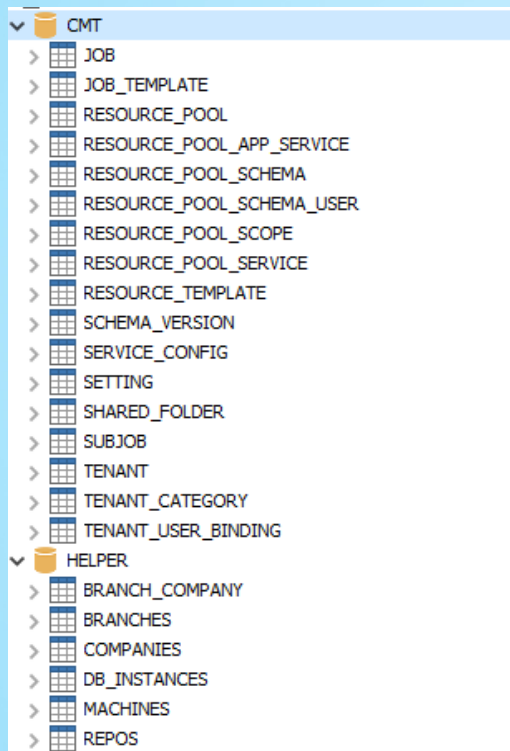
This build requires parameters:

REPO	<input type="text"/>
NAMESPACE	<input type="text"/>
ACTION	<ul style="list-style-type: none">full-installfull-installdelta-installmirrorupgradeslow-testmigratesync-imagedelete-namespacedelete-servicedelete-mirror
ENV	<input type="text"/>

Configuration – CMT



Configuration – CMT



Spinnaker

- ❑ 支持主流Cloud Provide以及K8s
- ❑ 支持自定义Cloud Provide
- ❑ 支持集成/监听Jenkins Job
- ❑ 支持监听Docker Registry

Q & A