**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**УНИВЕРСИТЕТ ИТМО**

**Факультет информационных технологий и программирования**

**Теория систем и системный анализ**

**Лабораторная работа №2**

Оценка структурной сложности информационной системы

**Выполнили студенты группы № М3300**:  
Балашов С.А.

Гринин В.Н.

Сметанина Д.Д.

**САНКТ-ПЕТЕРБУРГ**

**2019**

**Задание 1.** Определите число дуг в графе вашей системы, используя матрицу инциденций, построенную в ЛР 1

В соответствии с матрицей инциденций, число дуг графа равно S(G) = 18

**Задание 2.** Определите количество и структуру бикомпонент для вашей системы. Постройте для нее граф Герца. Дайте содержательное описание построенных артефактов

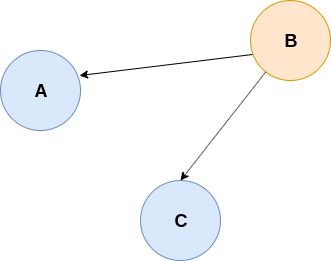


Рисунок 1. Граф Герца

Как видно из рисунка 1 система состоит из трех компонент сильной связности, в каждую из которых входит:

А:

* etcd.

B:

* kubectl;
* controller-manager;
* apiserver;
* scheduler;
* kubelet.

C:

* container(s);
* kube-proxy;
* ingress-controller;
* external client.

Что вполне логично, т.к. их назначение можно грубо описать как: А - хранение состояния, B - управление, C - взаимодействие между клиентом и приложением.

**Задание 3.** Постройте матрицу контуров для вашей системы, дайте ее содержательную интерпретацию

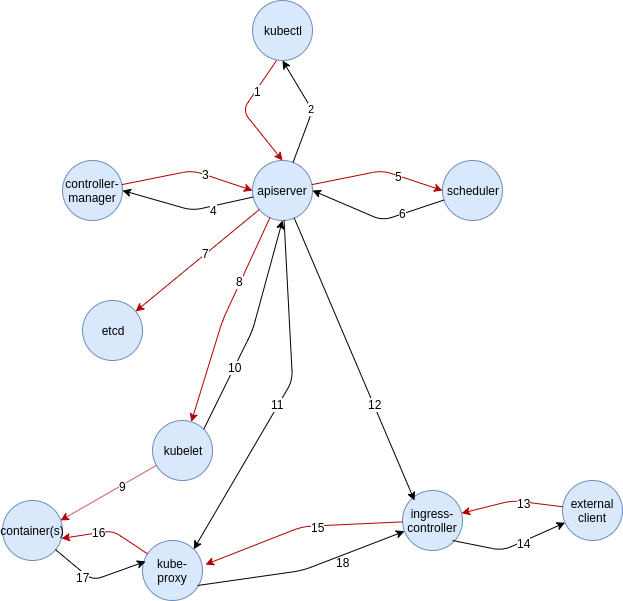


Рисунок 2. Все ветви и все ребра + остовное дерево

Таблица 1. Матрица контуров

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 3 | 5 | 7 | 8 | 9 | 16 | 15 | 13 |
| 2 | +1 |  |  |  |  |  |  |  |  |
| 4 |  | +1 |  |  |  |  |  |  |  |
| 6 |  |  | +1 |  |  |  |  |  |  |
| 10 |  |  |  |  | +1 |  |  |  |  |
| 17 |  |  |  |  |  |  | +1 |  |  |
| 11 |  |  |  |  | -1 | -1 | +1 |  |  |
| 18 |  |  |  |  |  |  |  | +1 |  |
| 12 |  |  |  |  | -1 | -1 | +1 | +1 |  |
| 14 |  |  |  |  |  |  |  |  | +1 |

В качестве остовного дерева мы взяли ребра 1, 3, 5, 7, 8, 9, 16, 15 и 13, поскольку выбор остовного дерева не несет смысловой нагрузки относительно связей в системе (он нужен нам только для расчета сложности графа). Остальные ребра остались хордами. Как видно, у ребра 7 нет значений в таблице. Это связано с тем, что между apiserver и etcd нет другого пути, кроме как ребро 7.

**Задание 4.** Перестройте полученную матрицу контуров, отсортировав ее строки по убыванию количества связей, входящих в контур. Дайте содержательное описание назначения каждого контура в информационной системе, инициатора (инициаторов) и конечных потребителей информации в каждом контуре, определите, являются ли однородными отдельные связи, входящие в контур (используют одинаковый способ связи, одинаковый или просто преобразуемый формат данных и т.п.).

Перестройте полученную матрицу контуров, отсортировав ее столбцы по убыванию количества контуров в которые входит соответствующая связь. Дайте содержательное описание различий в работе связи при ее использовании в различных контурах.

Таблица 2. Матрица контуров, отсортированная по строкам

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 3 | 5 | 7 | 8 | 9 | 16 | 15 | 13 |
| 12 |  |  |  |  | -1 | -1 | +1 | +1 |  |
| 11 |  |  |  |  | -1 | -1 | +1 |  |  |
| 2 | +1 |  |  |  |  |  |  |  |  |
| 4 |  | +1 |  |  |  |  |  |  |  |
| 6 |  |  | +1 |  |  |  |  |  |  |
| 10 |  |  |  |  | +1 |  |  |  |  |
| 17 |  |  |  |  |  |  | +1 |  |  |
| 18 |  |  |  |  |  |  |  | +1 |  |
| 14 |  |  |  |  |  |  |  |  | +1 |

Контуры:

12-8-9-16-15 - Маршрутизация трафика к сервисам

11-8-9-16 - Создание пода, запуск контейнера и маршрутизация трафика на нем

1-2 - Отправка запроса с командной строки и получение ответа

3-4 - Нотификация о новом ресурсе и внесение изменений в систему

5-6 - Нотификация о поде, для которого нужна привязка и привязывание пода к узлам

8-10 - Регистрация узла, на котором запущен kubelet и нотификация о поде, готовом к созданию

16-17 - Маршрутизация трафика к поду

15-18 - Маршрутизация трафика к нужному сервису и ответ ingress controller’у

13-14 - Взаимодействие с внешним пользователем

Итого 9 контуров.

Таблица 3. Матрица контуров, отсортированная по столбцам

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 8 | 16 | 9 | 15 | 1 | 3 | 5 | 13 | 7 |
| 12 | -1 | +1 | -1 | +1 |  |  |  |  |  |
| 11 | -1 | +1 | -1 |  |  |  |  |  |  |
| 2 |  |  |  |  | +1 |  |  |  |  |
| 4 |  |  |  |  |  | +1 |  |  |  |
| 6 |  |  |  |  |  |  | +1 |  |  |
| 10 | +1 |  |  |  |  |  |  |  |  |
| 17 |  | +1 |  |  |  |  |  |  |  |
| 18 |  |  |  | +1 |  |  |  |  |  |
| 14 |  |  |  |  |  |  |  | +1 |  |

При выборе в качестве остовного дерева другого набора ребер, мы фактически рассматриваем другой рабочий процесс системы, другой путь данных. Принципиальных различий в матрице контуров не будет, так как большинство контуров системы - два ребра между apiserver и другим компонентом системы, или между какими-то другими компонентами системы.

**Выводы:**

В процессе лабораторной работы была оценена структурная сложность информационной системы.

Так как наш граф невзвешен, то его сложность равна произведению числа дуг и числа контуров, то есть 18\*9 = 162.

Полученные характеристики и схемы полностью описывают графическое представление системы.