

АНАЛИЗ ИНФОРМАЦИОННОЙ СТРУКТУРЫ КОРПОРАТИВНОЙ СЕТИ

Рассмотрены проблемы анализа информационной структуры корпоративной сети. Сформулированы основные принципы анализа структуры. Решены задачи описания структуры и вычисления характеристик структуры: интенсивностей потоков данных между узлами сети, нагрузка на узлы и структурообразующее оборудование сети.

Введение

Структура корпоративной сети является основным фактором, влияющим на качество обмена данными между прикладным программным обеспечением корпоративной системы, и, соответственно, на качество решения прикладных задач. Анализ структуры является необходимым условием при выборе вариантов построения корпоративной сети и управления сетью [2, 3]. Результатами анализа должны стать численные значения характеристик сети: нагрузка каналов связи и структурообразующего оборудования, интенсивности потоков данных и запросов, поступающих на узлы сети [1, 2]. Указанные характеристики должны вычисляться с учетом специфических особенностей конкретной структуры сети.

1 Общие проблемы и принципы анализа структуры корпоративной сети

Проблемы анализа структуры корпоративной сети заключаются в том, что отсутствует единый подход к формированию структуры (в настоящее время существует, как минимум, несколько технологий проектирования сетей), имеется явная зависимость характеристик сети от параметров прикладных задач, решаемых в сетевой среде, отсутствуют отработанные математические методы формального описания структуры сети [2, 3].

Все это требует разработки принципов анализа структуры сети, инвариантных относительно технологии создания сети, решаемых прикладных задач, используемого для построения сети оборудования.

Для построения сети необходимо знать ее информационную структуру, которая определяет информационные потоки между узлами, на которых установлено программное обеспечение корпоративной системы. Под информационной структурой сети будем понимать совокупность информационных ресурсов корпоративной системы, размещенных на узлах сети, и информационных потоков между узлами, возникающих при решении прикладных задач. Узлом информационной структуры будем называть место размещения ресурса (приложения, базы данных), где обеспечивается его работа. Располагая данными об информационной структуре сети, можно принимать решения об организации каналов связи между узлами сети, определять необходимые параметры каналов связи и сетевого оборудования, формировать техническую структуру сети.

Под технической структурой будем понимать совокупность реального структурообразующего оборудования, узлов и каналов связи, составляющих полностью связанную сеть. Узлом технической структуры будем называть совокуп-

ность реальных технических средств, реализующих узел информационной структуры и обеспечивающих работу установленных ресурсов.

Для полноценного анализа структуры сети необходимо провести анализ ее информационной и технической структур, связать результаты анализа. Связывание результатов анализа информационной и технической структур подразумевает отображение параметров и характеристик информационной структуры в параметры технической структуры и определение параметров и характеристик технической структуры.

Схема проведения анализа структуры сети приведена на рис. 1.



Рис. 1 Схема анализа структуры корпоративной сети

2 Описание и расчет характеристик информационной структуры корпоративной сети

Отметим, что источниками потоков данных в сети являются пользователи и приложения. Пользователи инициируют работу приложений, которые в свою очередь обращаются к базам данных, к другим приложениям, передавая и получая различную информацию.

Работа приложений определяется задачами, которые решаются системой. Число задач в системе обозначим L .

Будем считать для упрощения, что каждое приложение, независимо от того, в какой задаче оно используется, работает всегда одинаково. Так что если в зависимости от задачи меняется работа приложения, то будем считать, что это уже другое приложение.

Под приложением будем понимать программу, которая запускается при решении задачи, программа может быть как специальной, написанной для решения задачи, так и общесистемной, предназначенной для выполнения стандартных процедур, которые требуются при решении задачи.

В каждой системе имеются хранилища данных, которые в нашем случае будем называть базами данных; их число в системе обозначим R . Считаем, что базы данных сформированы, состав данных определен и определены приложения, которые работают с базами данных.

Пусть число узлов сети M , количество пользователей системы N . Пусть в системе функционирует D различных приложений. Приложения предназначены для решения прикладных задач при обслуживании пользователей корпоративной системы.

Каждая задача номер k характеризуется следующим набором параметров: $\mathbf{S}_k = \{\mathbf{p}_k, \mathbf{d}_k, \mathbf{u}_k, \mathbf{W}_k\}$, ($k = 1, 2, \dots, L$).

Вектор-строка $\mathbf{p}_k = (p_{k1}, p_{k2}, \dots, p_{kD})$ определяет приложения, которые исполняются при выполнении задачи номер k , при этом $p_{ki} = 1$, если приложение номер i выполняется при решении задачи номер k , и $p_{ki} = 0$, если приложение номер i не выполняется при решении задачи номер k ($i = 1, 2, \dots, D$; $k = 1, 2, \dots, L$). Для всех элементов вектор-строки \mathbf{p}_k должно выполняться

условие $\sum_{i=1}^D p_{ki} \geq 1$, означающее, что при решении задачи номер k используется хотя бы одно приложение.

Из вектор-строк $\mathbf{p}_k = (p_{k1}, p_{k2}, \dots, p_{kD})$ можно составить матрицу $\mathbf{P} = \|\|p_{ki}\|$, задающую связи между задачами системы и приложениями.

Вектор-строка $\mathbf{d}_k = (d_{k1}, d_{k2}, \dots, d_{kR})$ определяет базы данных, которые используются при выполнении задачи номер k , при этом $d_{ki} = 1$, если база данных номер i используется при решении задачи номер k , и $d_{ki} = 0$, если база данных номер i не используется при решении задачи номер k ($i = 1, 2, \dots, R$; $k = 1, 2, \dots, L$). Для всех элементов вектор-строки \mathbf{d}_k должно выполняться условие

условие $\sum_{i=1}^R d_{ki} \geq 0$, означающее, что при решении задачи номер k может не исполь-

зоваться ни одной базы данных. Из вектор-строк $\mathbf{d}_k = (d_{k1}, d_{k2}, \dots, d_{kR})$ можно составить матрицу $\mathbf{D} = \|\|d_{ki}\|\|$, задающую связи между задачами системы и базами данных.

Вектор-строка $\mathbf{u}_k = (u_{k1}, u_{k2}, \dots, u_{kN})$ определяет множество пользователей системы, которым требуется запускать задачу номер k , при этом $u_{kj} = 1$, если пользователь номер j запускает задачу номер k , и $u_{kj} = 0$, если пользователь номер j не запускает задачу номер k ($j = 1, 2, \dots, N$; $k = 1, 2, \dots, L$). Для всех элементов вектор-строки \mathbf{u}_k должны выполняться условия:

$$1. \forall k = 1, 2, \dots, L; \sum_{j=1}^N u_{kj} \geq 1, \text{ т.е. каждая задача запускается хотя бы одним пользователем.}$$

$$2. \forall j = 1, 2, \dots, N; \sum_{k=1}^L u_{kj} \geq 1, \text{ т.е. каждый пользователь запускает хотя бы одну задачу.}$$

Из вектор-строк $\mathbf{u}_k = (u_{k1}, u_{k2}, \dots, u_{kN})$ можно составить матрицу $\mathbf{U} = \|\|u_{kj}\|\|$, определяющую потребности пользователей системы в запуске задач.

Матрица $\mathbf{W}_k = \|\|w_{kij}\|\|$, ($i = 1, 2, \dots, D$; $j = 1, 2, \dots, D$) определяет последовательность запуска приложений при решении задачи номер k . Так, $w_{kij} = 1$, если приложение номер j при решении задачи k запускается после окончания работы приложения номер i ; $w_{kij} = 0$, если приложение номер j при решении задачи k не запускается после окончания работы приложения номер i . Кроме того, $w_{kii} = 1$, если приложение номер i запускается первым при решении задачи k . Если приложение j при решении задачи k запускается последним, то

$$\sum_{m=1}^D w_{kjm} = 0 \text{ и } \sum_{m=1}^D w_{kmj} \geq 1.$$

Каждый i -й пользователь характеризуется интенсивностью потока запросов на запуск задач в системе – $\lambda_{ij} \geq 0$ ($i = 1, 2, \dots, N$; $j = 1, 2, \dots, L$). Здесь λ_{ij} – интенсивность потока запросов от пользователя номер i на запуск задачи номер j . Множество интенсивностей потоков запросов от пользователей на запуск задач будем задавать матрицей $\mathbf{\Lambda} = \|\|\lambda_{ij}\|\|$. Очевидно, что $\lambda_{ij} = 0$, если $u_{ji} = 0$, т.е. интенсивность потока запросов на запуск задачи j от пользователя i равна нулю, если этот пользователь не запускает задачу.

Каждое приложение номер m , используемое при решении задачи k , будем задавать набором: $\mathbf{A}_{km} = \{\mathbf{v}_{km}, \mathbf{b}_{km}\}$, ($k = 1, 2, \dots, L$; $m = 1, 2, \dots, D$). Здесь вектор-строка $\mathbf{v}_{km} = (v_{km1}, v_{km2}, \dots, v_{kmR})$ определяет объемы данных, которыми обменивается приложение m с базами данных за один сеанс решения задачи k . Так, $v_{kmr} \geq 0$ – объем данных, которыми обменивается приложение m с базой данных r . Из вектор-строк $\mathbf{v}_{km} = (v_{km1}, v_{km2}, \dots, v_{kmR})$ можно составить матри-

цы $\mathbf{V}_k = \|v_{kmr}\|$, ($m = 1, 2, \dots, D$; $r = 1, 2, \dots, R$), задающие объемы передаваемых данных между приложениями и базами данных при решении задач.

Вектор-строка $\mathbf{b}_{km} = (b_{km1}, b_{km2}, \dots, b_{kmD})$ определяет объемы данных, которыми обменивается приложение номер m с другими приложениями при решении задачи номер k . Так, $b_{kmj} \geq 0$ – объем данных, которыми обменивается приложение m с приложением j . Из вектор-строк $\mathbf{b}_{km} = (b_{km1}, b_{km2}, \dots, b_{kmD})$ можно составить матрицы $\mathbf{B}_k = \|b_{kmd}\|$, ($m = 1, 2, \dots, D$; $d = 1, 2, \dots, D$), задающие объемы передаваемых данных между приложениями.

Таким образом, интенсивности и объемы потоков данных в сети корпоративной системы определяются интенсивностями запуска задач пользователями и запускаемыми при этом приложениями. Здесь условимся, что объем передаваемых по сети данных задается в установленных единицах, например, байтах.

Зададим размещение приложений по узлам сети с помощью матрицы $\mathbf{G} = \|g_{ij}\|$, где $g_{ij} = 1$, если приложение номер i установлено на узле номер j , и $g_{ij} = 0$, если приложение номер i не установлено на узле номер j (здесь $i = 1, 2, \dots, D$; $j = 1, 2, \dots, M$). Для элементов матрицы \mathbf{G} должны выполняться условия:

1. Для любого $j = 1, 2, \dots, M$ справедливо неравенство $\sum_{i=1}^D g_{ij} \geq 1$, т.е. на

каждом узле должно быть установлено хотя бы одно приложение, но на узле может быть установлено и несколько приложений.

2. Для любого $i = 1, 2, \dots, D$ справедливо неравенство $\sum_{j=1}^M g_{ij} = 1$, т.е. ка-

ждое приложение должно быть установлено только на одном узле.

Из приведенных условий следует, что общее количество установленных

на узлах приложений равно общему количеству приложений $\sum_{i=1}^D \sum_{j=1}^M g_{ij} = D$.

Подключение пользователей к узлам задается матрицей $\mathbf{H} = \|h_{ij}\|$, где $h_{ij} = 1$, если пользователь номер i подключен к узлу номер j , и $h_{ij} = 0$, если пользователь номер j не подключен к узлу номер i ($i = 1, 2, \dots, N$; $j = 1, 2, \dots, M$). Для элементов матрицы \mathbf{H} выполняются условия:

1. $\sum_{i=1}^M \sum_{j=1}^N h_{ij} = M$, т.е. все пользователи системы должны быть подклю-

чены к узлам сети системы.

2. Для любого $i = 1, 2, \dots, N$ справедливо равенство $\sum_{j=1}^M h_{ij} = 1$, т.е. каж-

дый пользователь должен быть подключен только к одному узлу (этот узел является рабочим местом пользователя).

3. Для любого $j = 1, 2, \dots, M$ справедливо неравенство $0 \leq \sum_{i=1}^N h_{ij} \leq N$, т.е.

к любому узлу может быть либо подключено несколько пользователей, либо ни одного.

Распределение баз данных по узлам сети задается матрицей $\mathbf{S} = \|s_{rm}\|$, где $s_{rm} = 1$, если база данных номер r размещена на узле номер m , и $s_{rm} = 0$, если база данных номер r не размещена на узле сети номер m , ($r = 1, 2, \dots, R$; $m = 1, 2, \dots, M$). Для элементов матрицы должны выполняться следующие условия:

1. $\sum_{r=1}^R \sum_{m=1}^M s_{rm} = R$, т.е. все базы данных системы должны быть размещены на узлах сети системы.

2. Для любого $r = 1, 2, \dots, R$ справедливо равенство $\sum_{m=1}^M s_{rm} \geq 1$, т.е. каждая база данных может быть размещена более чем на одном узле сети, что на практике соответствует, например, репликации данных.

3. Для любого $m = 1, 2, \dots, M$ справедливо неравенство $\sum_{r=1}^R s_{rm} \geq 0$, т.е.

на любом узле может быть размещено либо несколько баз данных, либо ни одной.

Здесь распределенную базу данных будем рассматривать как совокупность отдельных баз данных.

Пример 1. Описание информационной структуры сети.

В качестве примера приведено описание информационной структуры для случая решения трех задач (задачи 1–3).

Считаем, что число пользователей системы $N = 6$, число узлов $M = 10$, число приложений $D = 7$, число баз данных $R = 3$. При этом одна база номер 3 используется двумя задачами. Пользователи 3 и 6 задачи не запускают.

Задачи 1, 2, 3 задаются следующими наборами:

$\mathbf{S}_1 = \{\mathbf{p}_1, \mathbf{d}_1, \mathbf{u}_1, \mathbf{W}_1\}$, где $\mathbf{p}_1 = (1, 1, 0, 0, 0, 0, 0)$, $\mathbf{d}_1 = (0, 1, 0)$, $\mathbf{u}_1 = (1, 0, 0, 0, 0, 0)$;

$\mathbf{S}_2 = \{\mathbf{p}_2, \mathbf{d}_2, \mathbf{u}_2, \mathbf{W}_2\}$, где $\mathbf{p}_2 = (0, 0, 0, 0, 1, 1, 0, 0)$, $\mathbf{d}_2 = (0, 0, 1)$, $\mathbf{u}_2 = (0, 0, 0, 1, 0, 0)$;

$\mathbf{S}_3 = \{\mathbf{p}_3, \mathbf{d}_3, \mathbf{u}_3, \mathbf{W}_3\}$, где $\mathbf{p}_3 = (0, 0, 1, 0, 0, 1, 1)$, $\mathbf{d}_3 = (1, 0, 1)$, $\mathbf{u}_3 = (0, 1, 0, 0, 1, 0)$;

$$\mathbf{W}_1 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad \mathbf{W}_2 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$

$$\mathbf{W}_3 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \end{pmatrix}.$$

Используемые приложения задаются наборами:

$$\mathbf{A}_{11} = \{\mathbf{v}_{11}, \mathbf{b}_{11}\}, \text{ где } \mathbf{v}_{11} = (0, 100, 0), \mathbf{b}_{11} = (0, 10, 0, 0, 0, 0, 0);$$

$$\mathbf{A}_{12} = \{\mathbf{v}_{12}, \mathbf{b}_{12}\}, \text{ где } \mathbf{v}_{12} = (0, 0, 0), \mathbf{b}_{12} = (5, 0, 0, 0, 0, 0, 0);$$

$$\mathbf{A}_{33} = \{\mathbf{v}_{33}, \mathbf{b}_{33}\}, \text{ где } \mathbf{v}_{33} = (50, 0, 50), \mathbf{b}_{33} = (0, 0, 0, 0, 0, 0, 10);$$

$$\mathbf{A}_{24} = \{\mathbf{v}_{24}, \mathbf{b}_{24}\}, \text{ где } \mathbf{v}_{24} = (0, 0, 100), \mathbf{b}_{24} = (0, 0, 0, 0, 10, 0, 0);$$

$$\mathbf{A}_{25} = \{\mathbf{v}_{25}, \mathbf{b}_{25}\}, \text{ где } \mathbf{v}_{25} = (0, 0, 0), \mathbf{b}_{25} = (0, 0, 0, 10, 0, 0, 0);$$

$$\mathbf{A}_{36} = \{\mathbf{v}_{36}, \mathbf{b}_{36}\}, \text{ где } \mathbf{v}_{36} = (0, 0, 0), \mathbf{b}_{36} = (0, 0, 20, 0, 0, 0, 0);$$

$$\mathbf{A}_{37} = \{\mathbf{v}_{37}, \mathbf{b}_{37}\}, \text{ где } \mathbf{v}_{37} = (0, 0, 0), \mathbf{b}_{37} = (0, 0, 0, 0, 0, 0, 0).$$

Считаем, что при запуске одного приложения другим возможен взаимный обмен данными. При этом объемы передаваемых данных задаются в одинаковых единицах измерения.

Матрицы \mathbf{H} и \mathbf{G} имеют вид

$$\mathbf{H} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \mathbf{G} = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

Матрицы \mathbf{S} и \mathbf{P} имеют вид

$$\mathbf{S} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}, \mathbf{P} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \end{pmatrix}.$$

Матрицы \mathbf{U} и \mathbf{D} имеют вид

$$\mathbf{U} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \end{pmatrix}, \mathbf{D} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 1 \end{pmatrix}.$$

На рис. 2 приведена информационная структура сети, соответствующей заданному описанию.

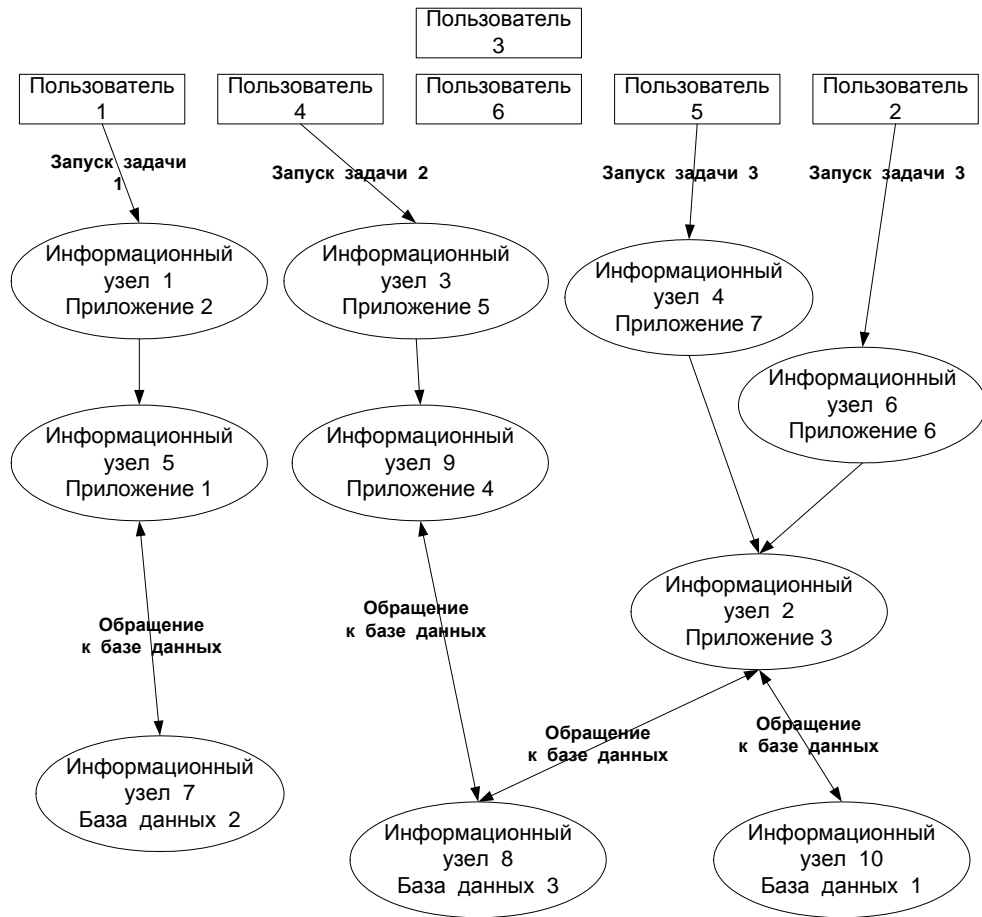


Рис. 2 Информационная структура сети

Таким образом, информационная структура сети корпоративной системы может быть задана набором следующих данных (параметров):

$$SI = \{N, M, D, L, R, S_i = \{p_i, d_i, u_i, W_i\} (i = 1, 2, \dots, L),$$

$$A_{km} = \{v_{km}, b_{km}\} (k = 1, 2, \dots, L; m = 1, 2, \dots, D), G, H, S\}.$$

3 Расчет параметров потоков данных между узлами информационной структуры сети

Используя введенные средства описания (определения, задания) информационной структуры сети в виде набора данных, можно определить параметры потоков данных между узлами сети.

Как уже отмечалось выше, источниками потоков данных являются запросы пользователей на запуск задач и запускаемые для решения задач приложения.

Значения элементов матрицы Λ интенсивностей потоков запросов пользователей на запуск задач определяются спецификой работы пользователей корпоративной сети, будем считать их известными.

Очевидно, что потоки запросов пользователей сначала поступают на те узлы сети, к которым прикреплены пользователи. Закрепление пользователей за узлами задается матрицей \mathbf{H} , определенной ранее.

Суммарная интенсивность потока запросов λ_k на запуск задачи k вычисляется по формуле

$$\lambda_k = \sum_{i=1}^N \lambda_{ik} \quad (k=1, 2, \dots, L). \quad (1)$$

Если ввести вектор-строку интенсивностей запуска задач в системе $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_L)$, то ее можно вычислить, используя (1), по формуле

$$\lambda = \mathbf{e}_N \mathbf{\Lambda}, \quad (2)$$

где \mathbf{e}_N – единичный вектор-строка размерности N .

Рассмотрим вектор-строку $\gamma = (\gamma_1, \gamma_2, \dots, \gamma_D)$:

$$\gamma = \lambda \mathbf{P}, \quad (3)$$

где элемент номер j вычисляется по формуле $\gamma_j = \sum_{k=1}^L \lambda_k p_{kj}$, что означает суммарную интенсивность запуска приложения номер j всеми задачами, решаемыми системой.

Таким образом, вектор-строка γ определяет интенсивности запуска приложений в корпоративной сети.

Определим суммарные объемы данных, передаваемых между узлами сети при решении задачи k . Пусть $\mathbf{Z}_k = \|z_{kij}\|$ – матрица, каждый элемент которой z_{kij} равен суммарному объему данных, передаваемому между узлами сети i и j при решении задачи k . Тогда можно записать, что

$$z_{kij} = \sum_{r=1}^D g_{ri} p_{kr} \left(\sum_{m=1}^D g_{mj} p_{km} b_{rm} \right) + \sum_{r=1}^D g_{ri} p_{kr} \left(\sum_{m=1}^R s_{mj} d_{km} v_{rm} \right), \quad (4)$$

$$k = 1, 2, \dots, L; \quad i = 1, 2, \dots, M; \quad j = 1, 2, \dots, M.$$

В формуле (4) первое слагаемое – объемы данных, передаваемых между всеми приложениями, используемыми задачей k и установленными на узлах i и j . Второе слагаемое – объемы данных, передаваемых между всеми приложениями и базами данных, используемыми задачей k и установленными на узлах i и j .

Используя (4) можно определить интенсивности потоков данных между узлами сети при решении задачи k . Естественно, что эти интенсивности определяются интенсивностью запуска задачи k всеми пользователями сети.

Здесь считаем, что каждое приложение, используемое при решении задачи, запускается один раз, хотя это ограничение легко обойти.

Матрица интенсивностей потоков данных между узлами сети при решении задачи k вычисляется по формуле

$$\mathbf{A}_k = \lambda_k \mathbf{Z}_k, \quad (k=1, 2, \dots, L). \quad (5)$$

Формулу (5) можно записать и так: $\mathbf{A}_k = \|\alpha_{kij}\|$, где $\alpha_{kij} = \lambda_k z_{kij}$ ($i = 1, 2, \dots, M; j = 1, 2, \dots, M$), α_{kij} – суммарная интенсивность потоков данных между узлами i и j при решении задачи k .

Теперь, используя (4), можно определить суммарные интенсивности потоков данных, передаваемых между информационными узлами. Эти интенсивности задаются матрицей

$$\mathbf{A} = \sum_{k=1}^L \mathbf{A}_k, \quad (6)$$

или $\mathbf{A} = \|\alpha_{ij}\|$, где $\alpha_{ij} = \sum_{k=1}^L \alpha_{kij} = \sum_{k=1}^L \lambda_k z_{kij}$ ($i = 1, 2, \dots, M; j = 1, 2, \dots, M$) – суммарная интенсивность потоков данных, передаваемых от узла i к узлу j .

Формулы (5) и (6) определяют загрузку каналов связи и коммуникационного оборудования сети.

Определим теперь нагрузку на информационные узлы. Будем исходить из того, что нагрузка на узел, на котором установлены приложения (базы данных) определяется интенсивностью потока запросов к этим приложениям (базам данных). Из формулы (3) известны интенсивности запуска приложений, установленных на узлах сети γ_j ($j = 1, 2, \dots, D$). Используя эти данные и данные о распределении приложений по узлам (матрица \mathbf{G}), получим формулу для вычисления интенсивности потока запросов на запуск приложения j , установленного на узле i :

$$\beta_{ji} = \gamma_j g_{ji}, \quad j = 1, 2, \dots, D; \quad i = 1, 2, \dots, M. \quad (7)$$

Это равенство можно для всех j и i записать в матричной форме в виде

$$\mathbf{B}^* = \|\beta_{ji}\| = \mathbf{\Gamma}_{dg} \mathbf{G},$$

где $\mathbf{\Gamma}_{dg} = \|\gamma_{ij}^*\|$ – диагональная матрица, у которой $\gamma_{ii}^* = \gamma_i$ и $\gamma_{ij}^* = 0$, если $i \neq j$, ($j = 1, 2, \dots, D; i = 1, 2, \dots, D$.)

Очевидно, что если $\beta_{ji} = 0$, то это значит, что приложение номер j не установлено на узле номер i . Формула (7) позволяет вычислить значение интенсивности потока запросов на запуск приложения от всех задач.

Для вычисления интенсивности потока запросов на запуск приложения номер j на узле i только от задачи k можно воспользоваться формулой

$$\beta_{kji} = \lambda_k g_{ji} p_{kj}, \quad k = 1, 2, \dots, L; \quad j = 1, 2, \dots, D; \quad i = 1, 2, \dots, M. \quad (8)$$

Из формул (8) и (7) следует, что $\beta_{ij} = \sum_{k=1}^L \beta_{kji}$.

Если на узле i установлена база данных, можно определить интенсивность потока запросов к базе данных j при решении задачи k :

$$\phi_{kji} = \lambda_k s_{ji} d_{kj}, \quad k = 1, 2, \dots, L; \quad j = 1, 2, \dots, R; \quad i = 1, 2, \dots, M. \quad (9)$$

Из выражения (9) получим формулу для вычисления суммарной интенсивности потока запросов к базе данных j , установленной на узле i при решении задач:

$$\varphi_{ji} = \sum_{k=1}^L \lambda_k s_{ji} d_{kj} = \sum_{k=1}^L \varphi_{kij}, \quad j=1, 2, \dots, R; \quad i=1, 2, \dots, M. \quad (10)$$

Здесь если $\varphi_{ji} = 0$, то на узле i не установлена база данных номер j .

Из величин φ_{ji} , вычисляемых по формуле (10), можно составить матрицу: $\Phi = \|\varphi_{ji}\|$, $j=1, 2, \dots, R; \quad i=1, 2, \dots, M$.

Отметим, что формулы (7) и (10) предусматривают также случаи, когда на одном узле установлено несколько приложений или несколько баз данных. Тогда суммарная интенсивность потока запросов на запуск приложений, установленных на узле i , вычисляется по формуле $\beta_i = \sum_{j=1}^D \beta_{ji}$, а суммарная ин-

тенсивность потока запросов к базам данных, установленным на узле i , вы-

числяется по формуле $\varphi_i = \sum_{j=1}^R \varphi_{ji}$.

Пример 2. Расчет параметров потоков данных для информационной структуры сети.

Рассчитаем параметры потоков данных для приведенного выше примера информационной структуры (рис. 1).

Матрица интенсивностей запросов пользователей на запуск задач имеет вид $\Lambda^T = \begin{bmatrix} 10 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 & 0 \\ 0 & 10 & 0 & 0 & 10 & 0 \end{bmatrix}$.

Из формул (1) и (2) имеем $\lambda = (10, 5, 20)$. Из формулы (3), используя матрицу \mathbf{P} примера 1, получим: $\gamma = (10, 10, 20, 5, 5, 20, 20)$.

Из выражения (4) можем вычислить матрицы $\mathbf{Z}_1, \mathbf{Z}_2, \mathbf{Z}_3, \mathbf{A}, \mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3$:

$$\mathbf{Z}_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 10 & 0 & 0 & 0 & 0 & 0 & 100 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix};$$

$$\mathbf{A}_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 50 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 50 & 0 & 0 & 0 & 0 & 500 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix};$$

$$\mathbf{A}_3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 200 & 0 & 0 & 0 & 1000 & 0 & 1000 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 400 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

Далее получим, используя (7) и (10):

$$\mathbf{B}^* = \begin{bmatrix} 0 & 0 & 0 & 0 & 10 & 0 & 0 & 0 & 0 & 0 \\ 10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 20 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 5 & 0 \\ 0 & 0 & 5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 20 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 20 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix};$$

$$\mathbf{\Phi} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 20 \\ 0 & 0 & 0 & 0 & 0 & 0 & 10 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 25 & 0 & 0 \end{bmatrix}.$$

Заключение

Предложенный подход к анализу структуры корпоративной сети и полученные результаты анализа информационной структуры позволяют администратору и разработчику сети оценивать загрузку каналов связи и сетевого оборудования при известных параметрах прикладных задач, решаемых на сети и структуре сети, сравнивать варианты структуры сети.

Список литературы

1. **Олифер, В. Г.** Новые технологии и оборудование IP-сетей / В. Г. Олифер, Н. А. Олифер. – СПб. : БХВ-Санкт-Петербург, 2000. – 512 с.
2. **Ретана, С.** Принципы проектирования корпоративных IP-сетей : пер. с англ. / С. Ретана, Д. Слайс, Р. Уайт. – М. : Вильямс, 2002. – 368 с.
3. **Таненбаум, Э.** Распределенные системы. Принципы и парадигмы / Э. Таненбаум, М. ван Стеен. – СПб. : Питер, 2003. – 877 с.