ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И ТЕЛЕКОММУНИКАЦИИ

УДК: 681.513

Мочалов Валерий Петрович, Ямбулатов Эдуард Искандарович

МОДЕЛЬ УПРАВЛЕНИЯ ВЫЗОВАМИ СИСТЕМЫ УПРАВЛЕНИЯ ТЕЛЕКОММУНИКАЦИЯМИ НА ОСНОВЕ ТЕХНОЛОГИИ CORBA

В статье приводится аналитическая модель для определения численных значений параметров системы управления вызовами удаленных методов и получения их оптимальных значений.

Ключевые слова: телекоммуникации, CORBA, программный компонент, процессорный модуль, объектный брокер, интеграция.

Mochalov Valerii P., Yambulatov Eduard I.

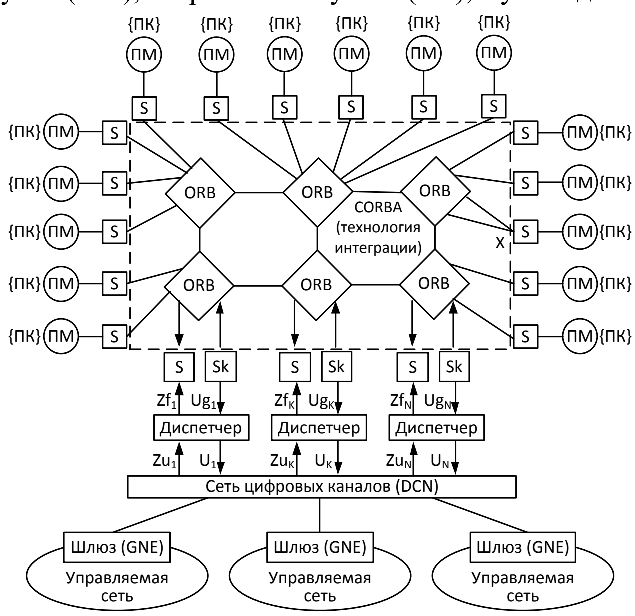
CALL MANAGEMENT MODEL IN TELECOMMUNICATIONS CONTROL SYSTEM BASED ON CORBA TECHNOLOGY

In this article we consider one of the elements of the decomposition process of integrating software components call management. We present a model to determine the numerical values of the call system management parameters of remote method and obtain their optimal values.

Key words: telecommunications, CORBA, a software component, a processor module, the object broker integration.

Существующие системы управления телекоммуникационными сетями и услугами (TMN, OSS/BSS и др.) имеют существенные недостатки в вопросе формирования новых телекоммуникационных услуг [1]. Устранить данные недостатки возможно путем построения распределенных систем управления (РСУ) на базе технологии CORBA.

Постановка задачи. Функциональная модель РСУ, представленная на рис. 1, состоит из N узлов процессорных модулей (ПМ), m транзитных узлов (ТУ), k узлов диспетчеров.



ПМ - процессорный модуль;
ORB – брокер объектных запросов;
Sk – скелетон;
S – клиентская заглушка;
Zf – запрос CORBA на формирование задания;
Ug – сформированное задание по ПК;
Zu – запрос на выполнение задания от пользователя;
U – выполнение задания;
{ПК} – множество программных компонент (ПК);
GNE – шлюз сетевого оборудования;
X – конфликт при обращении к одному и тому же ПМ при использовании одного и того де ПК различных заданий.

Рис. 1. Модель распределенной системы управления на основе CORBA

С сетевых элементов поступает поток входных сигналов, реакцией РСУ является множество выходных сигналов.

Качество функционирования РСУ оценивается набором значений параметров реализации всех процессов интеграции, представленных на рис. 2.

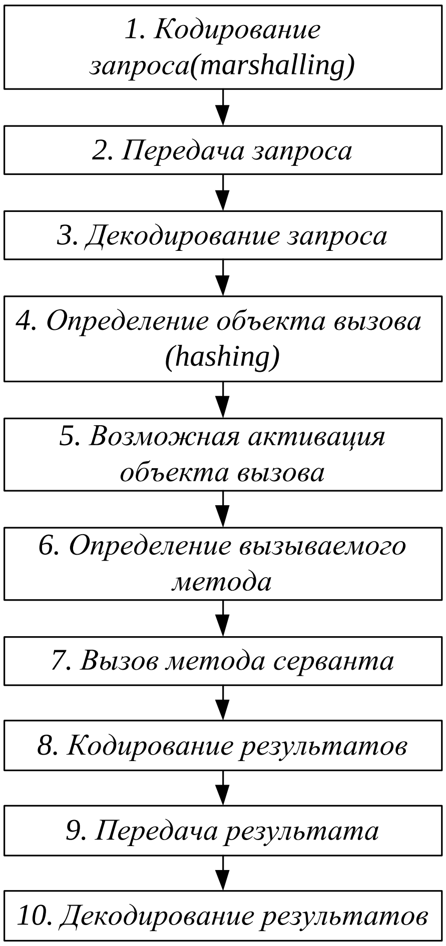


Рис. 2. Процессы выполнения запроса на интеграцию ПК

1. Кодирование (marshalling) запроса. Осуществляется упаковка и преобразование выходных данных для передачи по сети.
2. Передача запроса. Сформированный пакет передается адресату с помощью низлежащих сетевых сервисов.
3. Декодирование запроса. Получив пакет, данные декодируются параметры, имя метода и ключ объекта, необходимые для вызова метода.
4. Определение объекта вызова (hashing). Используя ключ объекта, сервер обращается к необходимому серванту.
5. Возможная активация объекта вызова. В случае отсутствия необходимого серванта его можно активировать при помощи запрограммированных правил.
6. Определение вызываемого метода. Машинный адрес необходимого метода определяется при помощи имени объекта.
7. Вызов метода серванта. Брокер осуществляет вызов необходимого метода серванта с обозначенными параметрами.
8. Кодирование результатов. Затем результат выполнения метода серванта вновь упаковывается для передачи в сеть и отправляется обратно с ответом.
9. Передача результата. Обратное сообщение использует то же соединение, которое было образовано для передачи запроса вызова метода.
10. Декодирование результатов. После получения пакета данные декодируются и передаются для дальнейшего использования.

Организационно-функциональная структура системы управления представлена на рис. 3.

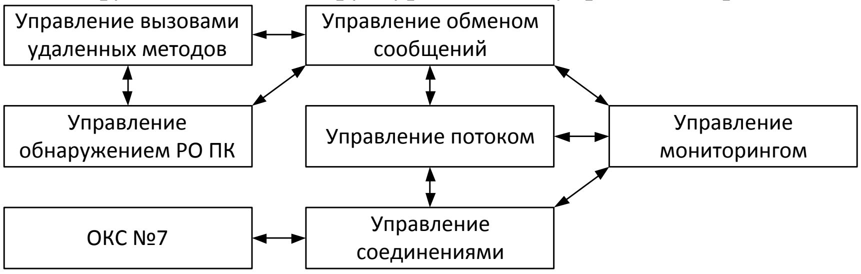


Рис. 3. Организационно-функциональная структура системы интеграции ПК

Исходя из логики работы, система интеграции ПК может быть разбита на следующие подсистемы:

- управление обменом сообщений;
- управление соединениями;
- ОКС №7;
- управление потоком;
- управление вызовами удалённых методов;
- управление мониторингом;
- управление обнаружением распределенных объектов ПК.

Процесс управления вызовами включает реализацию следующих процедур:

1) определяется физическое местонахождение в системе сервера, для которого предназначен данный вызов. Это шаг называется привязкой (binding) к серверу. Его результатом является адрес машины, на которую нужно передать вызов;

2) вызов процедуры и ее аргументы упаковываются в сообщение в формате, понятном серверной заглушке. Этот шаг называется маршалингом (marshaling);

3) полученное сообщение преобразуется в поток байтов (это сериализация, serialization) и отсылается с помощью какого-либо протокола, транспортного или более высокого уровня, на машину, на которой помещен серверный компонент;

4) после получения от сервера ответа, он распаковывается из сетевого сообщения и возвращается клиенту в качестве результата работы процедуры.

Для определения численных значений параметров системы управления вызовами удаленных методов и получения их оптимальных значений необходима разработка соответствующей модели.

Решение задачи. Функционально процесс управления вызовами можно представить в виде двухфазной СМО (рис. 4):

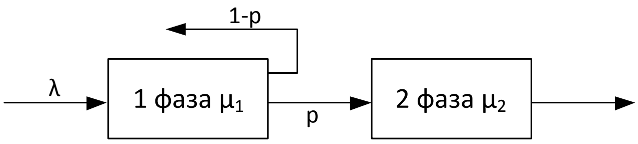


Рис. 4. Модель управления вызовами

- в первой фазе от *m* ЦУ имитируется поток запросов ПК $\{Z_{ПК}\}$ в систему интеграции РСУ;
- во второй фазе поступившие в систему интеграции запросы направляются в очередь для последующего обслуживания. В случае перегрузки очереди запрос отклоняется и отправляется уведомление серверу о необходимости повторного запроса через время, требуемое для обслуживания запроса, стоящего в очереди первым.

Запросы ПК на предварительной обработке проходят отбор при выполнении двух условий:

1) $\{Z_{ПК}\} \subset I$ (I – множество ПК реализуемых одним ПМ);

2) $\{Z_{ПК}\} \cap \{Z_{ПК}\} = \{Z_{ПК}\}$.

Первое условие позволяет разбить запросы, адресуемые ПК, отобрать поля параметров, маркировать значений полей параметров в соответствии с ЦУ, получать объектные ссылки взаимодействия РО. При это создается запрос $Z_{ПК[M\cup]}$, содержащий множество различных параметров.

Состояние данной системы опишем вектором $\{i,\ j\}$, где $i,\ j$ – количество вызовов на первой и второй фазах соответственно. Данный процесс является марковским $P(i,\ j,t)=P(i(t)=i,\ j(t)=j)$ и удовлетворяет равенствам [2]

$$P(i,\ j,t+\Delta t)=P(i,\ j,t)(1-\lambda\Delta t-i\mu_1\Delta t-j\mu_2\Delta t)+P(i+1,\ j-1,t)(i+1)\mu_1 r\Delta t+$$
$$+P(i+1,\ j,t)(i+1)\mu_1(1-r)\Delta t+P(i-1,\ j,t)\lambda\Delta t+P(i,\ j+1,t)(j+1)\mu_2\Delta t+o(\Delta t),$$

при условии $P(i,\ j,t_o)=P_o(i,\ j)$.

Очевидно, что уравнение для определения $P(i,\ j,t)$ будет иметь вид

$$\frac{\partial P(i,\ j,t)}{\partial t}+(\lambda+i\mu_1+j\mu_2)P(i,\ j,t)=\lambda P(i-1,\ j,t)+(i+1)\mu_1(1-r)\cdot$$
$$\cdot P(i+1,\ j,t)+(i+1)\mu_1 rP(i+1,\ j-1,t)+(j+1)\mu_2 P(i,\ j+1,t). \tag{1}$$

Решение уравнения (1) можно получить методом производящих функций ПФ

$$G(x,\ y,t)=\sum_{i,\ j=0}^{\infty}x^i y^i P(i,\ j,t). \tag{2}$$

С помощью дифференцирования ПФ можно вычислить моменты случайных величин i и j любого порядка, т. к. распределение любой случайной величины однозначно определяется ее ПФ.

Используя уравнение (1) и суммируя полученные равенства по всем i и j, находим

$$\sum_{i,\ j=0}^{\infty}x^i y^i\frac{\partial P(i,\ j,t)}{\partial t}+\lambda\sum_{i,\ j=0}^{\infty}x^i y^i P(i,\ j,t)+\mu_1\sum_{i,\ j=0}^{\infty}ix^i y^i P(i,\ j,t)+$$
$$+\mu_2\sum_{i,\ j=0}^{\infty}jx^i y^i P(i,\ j,t)=\lambda\sum_{i,\ j=0}^{\infty}x^i y^i P(i-1,\ j,t)+\mu_1(1-r)\sum_{i,\ j=0}^{\infty}(i+1)x^i y^i P(i+1,\ j,t)+ \tag{3}$$
$$+\mu_1 r\sum_{i,\ j=0}^{\infty}(i+1)^i y^i P(i+1,\ j-1,t)+\mu_2\sum_{i,\ j=0}^{\infty}(i+1)x^i y^i P(i,\ j+1,t).$$

Полученное дифференциальное уравнение в частных производных решаем методом представленном в [3]

$$G(x,\ y,t)=\phi(c_1,c_2)\exp\left\{\begin{aligned}&\frac{\lambda}{\mu_1}c_1(e^{\mu_1(t-t_o)}-1)-\\&-c_2\frac{\lambda\mu_1 r}{\mu_2-\mu_1}\left(\frac{e^{\mu_2(t-t_o)}-1}{\mu_2}-\frac{e^{\mu_1(t-t_o)}-1}{\mu_1}\right)\end{aligned}\right\}, \tag{4}$$

где $c_1=(x-1)e^{-\mu_1(t-t_o)}+(y-1)\dfrac{\mu_1 r}{\mu_2-\mu_1}(e^{-\mu_1(t-t_o)}-e^{-\mu_2(t-t_o)}),\qquad c_2=(y-1)e^{-\mu_2(t-t_o)}$,

$\phi(c_1,c_2)$ – произвольная дифференцируемая функция, определяемая из начальных условий.

Исходную производящую функцию обозначим

$$g(x,\ y)=\sum_{i,\ j=0}^{\infty}x^i y^i P(i,\ j,t),\qquad G(x,\ y,t_o)=g(x,\ y).$$

Из (4) получаем

$$G(x,\ y,t_o)=g(x,\ y)=\varphi(x-1,y-1). \tag{5}$$

$$G(x,\ y,t)=g(1+c_1,1+c_2)\times$$
$$\times xp\left\{\frac{\lambda}{\mu_1}c_1(e^{\mu_1(t-t_o)}-1)-c_2\frac{\lambda\mu_1 r}{\mu_2-\mu_1}\left(\frac{e^{\mu_2(t-t_o)}-1}{\mu_2}-\frac{e^{\mu_1(t-t_o)}-1}{\mu_1}\right)\right\}. \tag{6}$$

После преобразований имеем

$$G(x,y,t)=g\left\{\begin{matrix}1+(x-1)e^{-\mu_1(t-t_o)}+(y-1)\dfrac{\mu_1 r}{\mu_2-\mu_1}(e^{-\mu_1(t-t_o)}-e^{-\mu_2(t-t_o)}),1+\\ +(y-1)e^{-\mu_1(t-t_o)}\end{matrix}\right\}\times$$

$$\times\exp\left\{(x-1)\frac{\lambda}{\mu_1}(1-e^{-\mu_1(t-t_o)})\right\}\times \tag{7}$$

$$\times\exp\left\{(y-1)\frac{\lambda\mu_1 r}{\mu_2-\mu_1}(1-e^{-\mu_1(t-t_o)})-\frac{\mu_1}{\mu_2}\left(1-e^{-\mu_2(t-t_o)}\right)\right\}$$

При $t\to\infty$ производящая функция будет иметь вид

$$G(x,y)=g(1,1)\times\exp\left\{(x-1)\frac{\lambda}{\mu_1}\right\}\times\exp\left\{(y-1)\frac{\lambda r}{\mu_2-\mu_1}\left(1-\frac{\mu_1}{\mu_2}\right)\right\}=$$

$$=\exp\left\{(x-1)\frac{\lambda}{\mu_1}\right\}\times\exp\left\{(y-1)\frac{\lambda r}{\mu_2}\right\}, \tag{8}$$

из которого следует, что

$$P(i,\,j)=P_1(i)P_2(j)\,,$$

где

$$P_1(i)=\left(\frac{\lambda}{\mu_1}\right)^{\prime}\frac{1}{i!}e^{-\frac{\lambda}{\mu_1}},\;\; P_2(i)=\left(\frac{\lambda}{\mu_2}\right)^{\prime}\frac{1}{j!}e^{-\frac{\lambda}{\mu_2}}\,. \tag{9}$$

Равенства (9) позволяют определить число запросов на 1-й и 2-й фазах системы управления вызовами, учитывая, что

$$G(x,y,t)=\exp\left\{(x-1)\frac{\lambda}{\mu_1}(1-e^{-\mu_1(t-t_o)})\right\}\times$$

$$\times\exp\left\{(y-1)\frac{\lambda r}{\mu_2-\mu_1}(1-e^{-\mu_1(t-t_o)})-\frac{\mu_1}{\mu_2}\left(1-e^{-\mu_2(t-t_o)}\right)\right\} \tag{10}$$

$$P(i,\,j,t)=\left\{\frac{(\lambda a_1(t))'}{i!}e^{-\lambda a_1(t)}\right\}\left\{\frac{(\lambda a_2(t))'}{j!}e^{-\lambda a_2(t)}\right\} \tag{11}$$

где $a_1(t)=-\dfrac{1}{\mu_1}(1-e^{-\mu_1(t-t_o)})$,

$$a_2(t)=-\frac{r}{\mu_2-\mu_1}(1-e^{-\mu_1(t-t_o)})-\frac{\mu_1}{\mu_2}\left(1-e^{-\mu_2(t-t_o)}\right), \tag{12}$$

можно получить математическое ожидание и дисперсию данных случайных величин

$$M[i(t)]=D[i(t)]=\lambda a_i(t),$$
$$M[j(t)]=D[j(t)]=\lambda a_j(t). \tag{13}$$

Таким образом, используя полученные выражения, можно определять численные значения параметров системы управления вызовами удаленных методов, получать их оптимальные значения.

Литература

1. Мочалов В. П. Разработка распределенных систем управления телекоммуникационными сетями и услугами: дис. … д-ра техн. наук. Ставрополь, СевКавГТУ, 2006. 395 с.
2. Эльсгольц Л. Э. Дифференциальные уравнения и вариационное исчисление. М.: Наука, 1969. 424 с.
3. Саати Т. Элементы теории массового обслуживания. М.: Сов. Радио 1971. 570 с.