

## ПЕРЕДАЧА СООБЩЕНИЙ С ПРОМЕЖУТОЧНЫМ НАКОПЛЕНИЕМ

А.В. Николаев

(представил д.т.н., проф. В.И. Долгов)

Предлагается использование одного тракта для передачи сообщений, классифицированных по категориям срочности.

Как известно [1], в отличие от коммутации каналов, сети с коммутацией сообщений и коммутацией пакетов являются по сути своей маршрутизацией с промежуточным накоплением. Маршрутизация с промежуточным накоплением предполагает наличие очередей в узлах коммутации, которые возникают из-за неравномерности поступающей информации, вследствие необходимости сборки из мелких пакетов сообщений или более крупных блоков информации. Таким образом, при передаче возникают задержки. В зависимости от решаемой задачи задержки должны быть не более допустимой, минимальная величина которой изменяется от 1 до 10 секунд. В отличие от схемы передачи блоков данных в сеансе связи, при котором данные разной срочности передаются по разным трактам [2], предлагается схема, при которой передача осуществляется с использованием общего тракта. При этом выполняется селекция блоков разной срочности. Срочные блоки данных всегда идут впереди менее срочных.

Для этого используется два подхода, которые реализованы в разных информационно-вычислительных системах (ИВС). Суть предлагаемых подходов излагается ниже. Система «Сбор-В» и программно-аппаратные средства управления измерительными системами «Вега» из Единого центра управления, в процессе сеанса обмена блоками данных используют два класса срочности. К первому классу срочности относятся так называемые срочные данные. Они включают в себя в основном команды управления и квитанции на эти команды. Для них определена максимально допустимая задержка передачи. За пределами диапазона допуска задержки команда считается опоздавшей и потерявшей актуальность, а следовательно, должна удаляться из сети. Кроме срочных данных имеется также и второй класс сообщений – бессрочные данные. Они не должны теряться, но и не должны мешать прохождению срочных. Использование механизма кольцевого буфера на входе к точке доступа к услугам транспортного уровня позволяет выполнить виртуальное разделение

тракта передачи на тракт нормальных данных и тракт сигналов управления. На выходе из транспортного уровня кольцевой буфер размещается в точке, где сходятся потоки от разных источников информации.

Предлагаемая схема взаимодействия объектов сеансового уровня

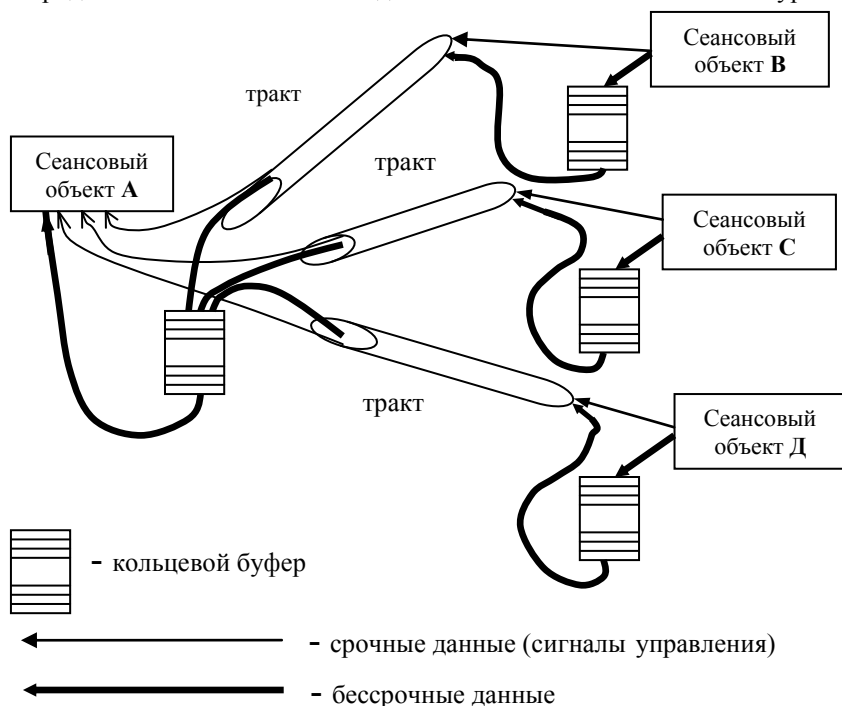


Рис. 1. Предлагаемый метод одностороннего взаимодействия сеансовых объектов

представлена на рис. 1. Кольцевой буфер образует очередь сообщений, построенную по принципу «первый пришёл - первый ушёл». Для улучшения понимания принципа работы буфера упростим ситуацию, предположив, что длина блоков одинаковая. Тогда работа буфера определяется двумя индексующими переменными: in - указывает позицию, куда пишется элемент, out - указывает позицию, из которой он извлекается. Идеально было бы, если у этого массива индексация не имела ограничений. Однако, вполне подходит и любой конечный массив, так как однажды считанный блок больше уже не используется. Поэтому его место можно повторно использовать. Кольцевой буфер используется для временного хранения бессрочных блоков данных до тех пор, пока не появится «окно» в потоке срочных данных. Глубина буфера должна быть такой, которая позволит при необходимости вместить «бессрочную» информацию одного сеанса. В связи с тем, что объём буфера может быть

значительным, его можно разместить в файле и хранить на дисковом накопителе. Это позволяет организовать последовательное группирование потоков срочных и «бессрочных» данных в общем тракте, отдавая преимущество срочным данным. Алгоритм работы процесса сортировки данных приведен на рис. 2. Здесь переменная  $n$  обозначает число элементов, находящихся в данный момент в буфере. Размер буфера обозначен величиной  $N$ . Под  $X$  понимаем содержимое сообщения.

```

ЦИКЛ-ПОКА идёт сеанс связи;
    ЦИКЛ-ПОКА есть сообщения в тракте;
        Читать сообщение из тракта;
        ЕСЛИ сообщение срочное;
            Передать в обработку сеансовому объекту  $A$ ;
        ИНАЧЕ
            Записать сообщение в кольцевой буфер;
        ВСЁ-ЕСЛИ
    ИНАЧЕ
        ЕСЛИ есть сообщение в кольцевом буфере  $n > 0$ ;
            Читать сообщение из кольцевого буфера;
            Передать в обработку сеансовому объекту  $A$ ;
        ВСЁ-ЕСЛИ
    ВСЁ-ПОКА
ВСЁ-ПОКА

ПРОЦЕДУРА Записать сообщение в кольцевой буфер;
    ЦИКЛ-ПОКА  $n < N$ ;
         $n = n + 1$ ;
         $buf[in] = X$ ;
         $in = in + 1$ ;
    ВСЁ-ПОКА;

ПРОЦЕДУРА Читать сообщение из кольцевого буфера;
    ЦИКЛ-ПОКА  $n > 0$ ;
         $n = n - 1$ ;
         $X = buf[out]$ ;
         $out = out + 1$ ;
    ВСЁ-ПОКА;

```

Рис. 2. Псевдокод процесса селекции срочных и бессрочных сообщений в сети

В гидрометеорологии определены 4 класса срочности. Первый – самый срочный, дальше номера в порядке уменьшения срочности. Процесс сбора метеоданных носит периодический характер. В связи с тем, что характер обмена данными в этой области человеческой деятельности

связан с так называемыми сроками сбора данных, то все задействованные линии связи загружены на сто процентов в периоды сбора данных. Информация не успевает передаваться в темпе поступления, что приводит к значительному накоплению её в очередях. Как же в этой ситуации развести в одном канале сообщения нескольких классов срочности? Для этого предлагается, во-первых, способ организации самих очередей и, во-вторых, способ обгона срочными сообщениями менее срочных.

Для того, чтобы не ограничивать размер очереди какой-либо определённой величиной, предлагается использовать списки, в которых в качестве указателя индекса служит значение, равное адресу этого сообщения в памяти ЭВМ. Если в узле коммутации применить в качестве операционной системы (ОС) UNIX, то можно организовать виртуальную память для хранения этих списков и выделения адресов не из реальной оперативной памяти, а из виртуальной памяти, которая значительно больше реальной оперативной и зависит от размера памяти на диске, отведенного под раздел свопинга. Таким образом, мы можем выделить для очередей значительное пространство, не заботясь о том, где реально оно будет находиться. При малых размерах списков, допускающих использование свободной оперативной памяти, очереди будут располагаться в ней. При превышении размеров свободной оперативной памяти, очереди будут дополнительно размещаться на диске. Управление этим размещением принимает на себя сама ОС UNIX. Для того, чтобы очередь предстала перед нами сегментированной по уровням срочности, следует использовать принцип, применённый в реляционных СУБД. А именно, следует сообщения и регистрационную информацию, характеризующую её, рассматривать как строки таблицы сообщений БД. После этого все сообщения разделить так, как это делается при помощи «взгляда» на несколько таблиц с одинаковой в них срочностью сообщений. Теперь дальнейшее продвижение по сети следует производить путём выдачи в тракт передачи по принципу «первый пришёл - первый ушёл» вначале всех сообщений первого класса срочности, затем - для второго и т. д., до тех пор, пока не выдастся самый низкоприоритетный класс.

Таким образом, предложенные подходы решения проблем селекции сообщений по классам срочности в реализации передачи с промежуточным накоплением позволяют использовать общий тракт сети передачи данных для повышения его эффективности.

## ЛИТЕРАТУРА

1. Бертсекас Д., Галлагер Р. Сети передачи данных. – М.: Мир, 1989. – 544 с.
  2. Якубайтис Э.Я. Архитектура вычислительных сетей. – М.: Статистика, 1980. – 280 с.
-