发明名称：分布处理系统中处理数据用的方法与装置

摘要
在一个分布系统中，包含通过一个传输介质相连的一组处理器和一种对可以在该传输介质上被再定位的可变长度数据的处理方法，其中各处理器利用表示出从传输介质接收到的数据的长度以及在处理器中所存数据的长度的信息而将从传输介质接收的数据存入各自的存储器的某个空白区中。
1. 在一个分布系统中，包含通过一个数据域互相连接的一组处理器和一种处理数据的方法，其中，各个处理器处理在该数据域上其长度可变的数据，并执行一个为存储从所述数据域接收到的数据而在一个存储器中查找所需的某个空白区的步骤，执行该步骤所根据的是指示出所述数据长度和在所述存储器中的所存数据的位置的信息。此外，根据查找结果再执行一个在所述存储器的空白区中存储从所述数据域中接收到的数据的步骤。

2. 根据权利要求1的一种处理数据的方法，其中各处理器执行一个对在所述存储器中所存数据进行再定位的步骤。

3. 根据权利要求2的一种处理数据的方法，其中各处理器利用指示出在所述存储器中所存数据的首地址的信息而访问这些数据，并运行所数据。

4. 根据权利要求1至3项中的任何一项的一种处理数据的方法，其中指示出上述所存数据的位置的信息包含一个识别码和一个空白区的长度。

5. 根据权利要求4的一种处理数据的方法，其中所识别码包含一个内容码。
说明

分布式处理系统中处理数据用的方法与装置

本发明系涉及在包括通过某种传输介质相联的一组处理器的分布式处理系统中作在线数据处理用的一种方法和装置。

关于在一个分布式处理系统中用作在线数据处理用的一个系统，在公开号为NO.146361/1982的日本专利中提出了一个系统，与一个处理器相联的某个传输控制处理器将该处理器所需的数据（以其内容码作为关键码）发送给一个传输介质。该处理器中的程序将所发送的数据存入数据区。但是迄今为止数据区的规模与首地址一直是在程序的准备阶段专门予以确定的。

因此，如果由程序直接存贮长度有所增加的数据，就会破坏直接位于该数据后的其它一些区域，除非采取下列措施：

（1）当通过程序的首地址直接访问数据区时，仅对数据区进行修正是不够的。因此，用于开发程序的处理器必须再次将数据区和访问该数据的程序相连，还必须将它们同时装入一个在离线状态下需要它们的一个处理器中。

（2）为实现上述联接关系，开发程序用的处理器对专门为接收程序的处理器用的存储器信息进行管理，并且对程序与数据进行管理。

（3）开发程序用的处理器确定一个装入地址，从而在连接时的地址与装入时的地址相符。

综上所述尽管在技术上，对于动态地解决传输介质上变长度数据的问题并未予以重视，因而产生如下所述的问题：
（1）如果由程序直接对长度有所增加的数据进行存储，就将破坏直接位于该数据后的其它一些区域，因而不能访问它们。
（2）数据不是在在线条件下存入的，并且必须修正程序。
（3）开发程序用的处理器必须对专门为处理器用的存贮器信息进行管理，还必须对程序和与开发及修正程序所必须的数据有关的信息进行管理，并且还必须对确定处理器用的装入地址进行管理。

由于这些原因，处理器如果只依靠它们的内部信息，而不向其它的处理器发出指令，或者不从其它处理器得到指令，是不能保持处理的自主权的。这就难以保持在维护与扩充方面的灵活性。

本发明的目的是提供处理数据的一种方法和一种装置，其中在传输介质上可被再定位的长度可变的数据是以某种在线方式通过查找并确保合适的空白区独立地被存入的，以清除目前存在的在线数据处理系统中所固有的上述问题。

为了达到上述目的，本发明提出一种处理数据的方法，其中，在一个包含通过某种传输介质互相连接的一种处理机的分布处理系统中，各个处理器对可以在数据区被再定位的变长度数据进行处理，其改进是：所述各个处理器具有指示出在其各自的存贮器中所存数据的位置的信息，查找为在存贮器中存贮数据所必需的区域的信息，以及只利用表示存于本处理器中的数据的位置的信息以间接访问的形式在线地将从传输介质接收到的数据存在各自存贮器的空白区中的信息。

对各图的简要说明如下：

图1是根据本发明的一个实施例说明在一个分布处理系统中的一个处理器的处理过程的流程图；

图2是说明本发明所涉及的分布处理系统的总结构的框图；

图3是说明图2中处理器的结构的框图；
图4表示了图3中一个存储器信息存储单元的格式；
图5表示了一个存储器分配图格式；
图6表示了根据本发明的另一个实例的处理器的结构；
图7表示了一个存储器分配图格式；
图8是一个流程图，表示了根据本发明的另一个实例在分布式处理系统中的处理器的处理过程；
图9表示了在利用本发明的一个数据驱动系统中所采用的处理器的结构；
图10是表示该数据驱动系统中的处理过程的流程图；
图11表示了图9中的连接信息存储单元的一种格式。

现在阐述本发明的原理。

根据本发明，包含通过一个传输控制处理器相连接的一组处理器的一个分布系统，其中流经一个传输介质的数据包含：表示其内容的一个内容码、识别数据体的一个识别码、数据体长度、以及数据体。对于各处理器所需的数据，其内容码已予先寄存在传输控制处理器中。

各处理器含有一个暂存输入数据用的缓冲器、一个存储数据作用的存储器存储单元、和一个存储器管理单元，它管理一个存储器信息存储单元，存储器信息存储单元为存储在数据存储单元中的数据，存储识别码、数据区长度和首地址，该存储器管理单元还管理处理器中的存储器。它还查找并确保一个合适的空白区。

由各处理器构成的数据以上述形式在传输介质上传输，传输控制处理器向处理器发送在处理器中的存储器管理单元所需的参数，以其内容码作为关键码。处理器根据输入到缓冲器的数据驱动存储器管理单元。存储器管理单元首先根据存储在存储器信息存储单元内的信息
计算数据区的首／尾地址。然后存储器管理单元根据它们和存贮在数据存贮单元内的上／下限地址（内部数据），在数据存贮单元上准备一张存储器分配图。

下一步，存储器管理单元根据已准备好的存储器分配图查找一个其长度大于所接收数据的长度的空白区。当存在一个合适的空白区时，与所接收的数据有关的信息就被存入存储器信息存贮单元的一个空白区内。然后存储器管理单元在数据存贮单元内存入数据体，以数据存贮单元中空白区的首地址作为装入地址。这样，由于处理器执行存储器管理并查找和确保一个合适的空白区，其它的区域就不被破坏，甚至当输入数据的长度变化时也如此。此外，由于没有必要修正存贮数据用的程序，就允许以在线方式存贮数据。还有，各个处理器不必识别专门为其它处理器用的信息，并能独立地进行处理。

现在结合附图详细地阐述本发明的实施例。

图2是用来说明根据本发明的一个实施例的某一个分布处理系统的总体结构的框图。该分布处理系统包含一组通过传输控制处理器（TCP）13而由一个公共传输介质（以下称为数据域）11联接起来的处理器14A。这里涉及的公共传输介质代表一种处理器的数据在其中流动的介质，诸如导线式传输通道、存贮器或无线通道。

现在参考图3说明处理器所进行的处理过程。数据12以上面所述的方式在数据域11内流动。这里数据体包含其长度可以重新排列的可变长度数据，例如普通数据、原程序或primitive program目标程序以及可以重新排列的装入模块。

现在阐述通过数据域11在各处理器14A问发送和接收数据的一种方法协议。这是一种利用称之为内容码的编码以表示数据内容的方法。当然允许采用任何其它的协议作为一种发送和接收数据的方法。
数据域1和12上的数据12是通过与各处理器14A相连的各NCP13接收到的。关于各处理器14A所需的参数，如上所述，其内容码以存在各NCP处。各NCP13以内容码作为关键字确定接收的数据，并根据需要将它们送给各处理器14A。

处理器14A中的程序是如此编制的：当具备了预定的内容码的数据时，便以该数据作为输入而执行处理过程。在前述公开号为NO.146361/1982的日本专利中对此已作了详细叙述。

处理器14A对输入到缓冲器18A处的数据12产生响应而驱动存储器管理单元15A。根据图4所示的存储器信息存储单元16A中的信息，存储器管理单元15A在后面将会阐述的过程中查找并确保一个合适的空白区，并在数据存贮单元17A中存入输入数据12。存储器管理单元15A包含一个存储器分配图构成单元15A、一个空白区查找单元15B、一个存储器信息形成单元15C和一个数据写入单元15D。

图1是一个流程图，按照该图所进行的过程是由存储器管理单元15A执行的。现在参考图1的流程图说明存储器管理单元15A的处理过程。

存储器管理单元15A首先驱动构成存储器分配图41A的存储器分配图构成单元15A，以便在数据存贮单元17A中查找空白区。图5表示了存储器分配图41A的一种格式。在存储器信息存贮单元16A中以图4所示的格式存贮着与已经存贮在数据存贮单元17A中的数据有关的信息。

因此，在准备存储器分配图41A时，检查是否使用存储器信息。亦即，当输入数据12的识别码ID与存储器信息存贮单元16A所指的数据的识别符相同则，就由存储器信息存贮单元16A所指
的数据进行重写刷新或如在步骤58中相应输入数据12，因此，在这种情况下，不利用存贮器信息（步骤51），过程进入步骤53。

当识别码不相同时，便以图5的格式根据存贮器信息构成存贮器分配图41A。这里，尾地址是通过将长度与首地址相加得到的。从存贮器分配图41A的第二行开始紧密地存放各识别符、首地址和尾地址组成的多组参数（步骤52）。步骤51和52的执行过程对所有由存贮器信息存贮单元16A所指定的数据都是有效的（步骤53）。

存贮器分配图41A的第一行和存放最后一个数据的下一行即第n行与数据存贮单元17A的上／下限地址（低／高）相符。因此，例如-1的一个值存在识别码中，并在首／尾地址中存入相同的值，如图5所示。然后，次序递增地存入存贮器分配图中第一行至第n行，以首地址作为关键码。为存贮器分配图41A中的第一行至第(n-1)行找到一个空白区的长度。这里分别用fc1、si、ei和di表示第i行空白区的识别码、首地址、尾地址和长度，紧接在识别码fc1的数据区后的空白区长度di为di = si+1- ei。第n行空白区的长度例如予定为-1（步骤54）。

然后存贮器管理单元15A驱动空白区查找单元15B，后者根据输入数据12的长度和存贮器分配图41A查找其长度大于输入数据12的长度的空白区。如果存在这样的空白区，就得到它的一个首地址，亦即是紧接在空白区前面的数据区的尾地址。然后，存贮器管理单元15A驱动存贮器信息构成单元15C。如果没有这样的空白区，便结束处理过程。

存贮器信息构成单元15C首先检查在存贮器信息存贮单元16A中是否存在一个用来寄存与输入数据12有关的存贮器信息的空白区。
其删除识别码与输入数据 12 的识别码相符合的所有数据，并将存储器信息存储单元 16A 中的区域与空白区相加（步骤 56）。如果存在这样的一个空白区，就在存储器信息存储单元 16A 中为输入数据 12 存入识别码。数据区大小、以及首地址。存储器管理单元 15A 驱动数据写入单元 15A（步骤 57）。

数据写入单元 15A 在数据存储单元 17A 的空白区中以首地址为装入地址存入输入数据 12 的数据体。

进面，作为一个有改动的实施例，内容码可以与识别码相符，因此基于识别码的处理过程可以利用内容码进行。

按照上述实施例，处理器管理存储器，并进面查找和确保合适的空白区，因此，即便输入数据的长度在动态地有所变化，数据存储单元上的其它区域并不被破坏。此外，由于不必对存储输入数据的程序进行修正，就允许以在线方式存入数据。还有，各处理器不必识别专门数据为其它处理器用的信息，并能独立地执行处理过程。

因此，允许各处理器在查找和确保一个合适的空白区时独立地以在线方式存入数据，因此系统的优点是改善了在线维护与扩充的能力。

图 6 示出了本发明的另一个实施例，其中的处理过程是由各处理器执行的。在图 6 中，标号 11、12 和 13 所示的结构单元与前述实施例中相同，而标号 14B 至 18B 所示的则与上述实施例中 14A 至 18A 所示的单元相同。即，标号 14B 表示一个处理器，标号 15B、16B、17B 和 18B 分别表示一个存储器管理单元、一个存储器信息存储单元、一个数据存储单元和一个缓冲器，它们组成处理器 14B。处理器 14B 中的程序是如此编制的，即当具有内容码的数据都备齐时，便以数据作为输入执行处理过程。这是与前述实施例相同的。
处理器14B对输入缓冲器18B的数据12作出响应，驱动存储器管理单元15B。根据图4所示的存储器信息存储单元16B中的信息，存储器管理单元15B通过下面叙述的过程确保一个合适的空白区，并将输入数据12存在数据存储单元17B中。存储器管理单元15B包含一个存储器分配构成单元15a、一个空白区查找单元15b、一个存储器信息构成单元15c、一个数据写入单元15d，以及一个再定位单元15e。

图8是说明存储器管理单元15b所执行的处理过程的流程图。现在结合图8的流程图说明按照这个实施例由存储器管理单元所执行的处理过程。

存储器管理单元15B首先驱动存储器分配构成单元15a，后
者构成一专用来如在前述实施例中所述那样，在数据存储单元17B中查找空白区的存储器分配图41B。图7表示了存储器分配图的一个格式。在存储器信息存储单元16B中按照图4所示的格式存入了与已经存储在数据存储单元17B中的数据有关的信息。

因此，在准备存储器分配图41B时，检查是否使用存储器信息。亦即，当数据输入12的识别码与由存储器信息存储单元16B所指定的数据的识别码相同时，存储器信息存储单元所指定的数据可以重写刷新或者在步骤70中响应输入数据12。在这种情况下，不利用存储器信息（步骤61）而使处理过程转入步骤63。

当两个识别码不相同时，则根据存储器信息按照图7的格式构成存储器分配图41B。

步骤64与65的操作内容与前述实施例中的步骤54和55的内容相同。如果不存在空白区，则存储器管理单元15B驱动再定位单元15e将所存数据重新定位以便检查是否获得所需的空白区。
假设再定位单元 15e 对所存数据作了再定位以便获得一个所需的空白区。首先，在所存数据以顺序增加等级字列出在数据存储单元 17B 上的情况，找到为每个区域用的存贮器分配图 41B，各首地址作为关键字。在存贮器分配图 41B 中，分别用 si、ei、di 和 osi 表示第 i 行的首地址、尾地址、空白区的长度以及再定位前的首地址，并以如下方式刷新：

\[
osi = si \\
si = ei - 1 \\
ei = ei - di - 1
\]

（式中 \(di - 1 \neq 0, i = 2, 3, \ldots, n - 1\)）

在构成存贮器分配图时，再定位前的首地址 osi 的初始值例如假定为 -1。

从而，对再定位前的首地址刷新，后面将叙述的存贮器信息构成单元 15c 和数据写入单元 15d 检查它是否是初始值，从而确定数据是否需要再定位，并识别了初始首地址。

下一步，找到空白区的长度并如同存贮器分配图构成单元 15a 的方式构成存贮器分配图 41B（步骤 66）。根据重构的分配图 41B，如同空白区查找单元 15b 的方式查找空白区。如果存在空白区，在存贮器管理单元 15b 便驱动存贮器信息构成单元 15c，从而与输入数据 12 和需要再定位的数据信息有关的信息被寄存到存贮单元 16B。如果没有这样的空白区，过程便结束（步骤 67）。

存贮器信息构成单元 15c 的工作与前述实施例中所述的相同（步骤 68）。这里有，如果存在空白区，在存贮器信息存贮单元 16B 中为输入数据 16 寄存识别码、数据区长度和首地址。上述的首地址是由空白区查找单元 15b 或再定位单元 15e 得到的一个空白区的
首先地址进位，如果存在需要再定位的存贮数据，则以识别码作为关键码查找存贮器信息存贮单元 16 B 的内容。区域的首地址被重构的存贮器分配图 41 B 中的首地址更新。然后，存贮器管理单元 15 B 驱动数据写入单元 15 d（步骤 69）。

数据写入单元 15 d 将输入数据 12 写到数据存贮单元 17 B 上并对需要再定位的存贮数据再定为。亦即，当存在需要再定位的存贮数据时，便根据存贮器分配图 41 B 中的信息对数据存贮单元 17 B 中的数据再定位。

然后，根据数据存贮单元 17 B 上的空白区的首地址存入输入数据 12 的数据体（步骤 70）。

上述实施例可以按下述方式改动：

（1）存贮器分配图 41 B 的识别码可以用作存贮器信息存贮单元 16 B 上的行号，这样可以由存贮器信息构成单元 15 c 对需要再定为的存贮数据进行高速处理。然而这里有不包括存贮器分配图 41 B 的第 1 行和第 n 行。这就可以简化利用识别码作为查找存贮器信息存贮单元 16 B 的内容的过程。

（2）可以使内容码与识别码相符，以便利用内容码进行根据识别码的处理过程。

根据上述实施例，处理器管理存贮器、查找和确保一个合适的空白区，并根据需要将存贮数据再定为。因此，防止了空白区进一步被分为小区域，并消除了浪费的空白区。这就在上述实施例所获得的效果之外得到可贮存更多的数据的特点。

因此，处理器在确保一个合适的空白区的同时以在线方式独立地存贮数据，从而可以有效地利用存贮器，改进了系统在线维护和扩充方面的性能。
下面叙述的是利用一个间接访问系统对由上述系统所存贮的数据进行处理的工作原理。

这个数据驱动系统的特点在于：处理器具备前述的适用于训练相互独立、互不干扰并可再定位的变长度数据的数据写入装置；特点还在于：处理器具有指示出所存数据首地址的信号；特点还在于处理器利用上述信息通过间接访问系统，执行与所存数据有关的处理过程。

下面将更具体地对此进行叙述。

图9具体示出了适合于图2的分布处理系统的数据驱动系统的情况，其中标号11、12和13所表示的结构与前面所指出的相同，标号14c至18c所表示的与14a至18a所表示的结构单元相同。亦即，标号14c表示一个处理器，标号15c、16c、17c和18c分别表示一个存储器管理单元、一个存储器信息存贮单元、一个数据存贮单元和一个缓冲器，它们组成了处理器。

处理器14c还具有一个在线装入器21、一个连接信息存贮单元22和一个执行管理单元23。在线装入器包含存储器管理单元15c和一个在线连接器21a。此外，存储器管理单元15c可以由存储器管理单元15a或是在前述实施例中所述的存储器管理单元15b构成，在本例中是后者。

处理器14c查找一个合适的空白区，存入输入数据12，访问所存数据，并执行处理过程。然而，在这里当所存数据包含一个可以再定位的输入模块并当上述所需数据作为输入时，就运行所存数据。

图10是处理器14c所执行的处理过程的流程图。现结合图9和图10叙述由处理器14c所执行的处理过程。

处理器14c对输入到缓冲器18c的数据12产生响应，驱动在线装入器21。
为了在存储器单元 17c 上查找空白区，在线装入器 21 驱动存储器管理单元 15c。后者根据存储器信息存储单元 16c 和数据存储单元 17c 中的上/下限地址（内部数据）构成与图 4 中所示的相同的寄存器分配图（步骤 71）。然后，根据存储器分配图查找并确保一个其长度大于输入数据 12 的长度的空白区。此时，所存储数据可就根据需要再定位，以消除浪费的空白区并查找和确保所需的空白区（步骤 72）。

下进一步，如果存在输入数据 12 和需要再定位的所存储数据，与此有关的信息则寄存到存储器信息存储单元 16c 或被刷新。于是，需要再定位的所存储数据被重新定位，而输入数据 12 便被存入数据存储单元 17c 中（步骤 73）。

下进一步，在线装入器 21 驱动在线连接器 21a，后者管理连接信息存储单元 22，然后处理器 14c 利用间接访问系统访问所存储数据并对它进行处理。图 11 显示了连接信息存储单元 22 中的一个格式。连接信息存储单元 22 的行数与存储器信息存储单元 16c 的相符。在线连接器 21a 顺序地为所有存储在数据存储单元 17c 中的数据从存储器信息存储单元 16c 中找到首地址和内容码，并刷新连接信息存储单元 22（步骤 74）。

执行管理单元 23 根据间接访问系统和用内容码作为关键码为需要输入的所存储数据查找连接信息存储单元 22，并读出首地址，访问所存储数据并运行之（步骤 75）。被运行的所存储的数据处理缓冲器 18c 中的输入数据，将内容码加到结果上并将它输出到数据域上。

上述实施例可以改行如下：

(1) 为了维护与刷新连接信息存储单元 22 有关的步骤，可以将它包括在存储器信息存储单元。
格和提高在线设备器 21 的处理速度。

（2）可以使内容码与识别码相符，根据识别码进行的处理过程可以利用内容码进行。

（3）存储器管理单元 15c 可以不具备再定位功能（在第一个实施例中所指出的那种功能）。

按照上述数据驱动系统，处理器具备能被再定位的存储器的控制功能、连接功能和执行管理功能，并查找一个合适的空白区，存储输入数据，访问所存数据并将其运行，这就改善了系统的在线维护和扩充能力。

按照上述本发明，各存储器利用指示出在各自的存储器中所存数据的位置的信息，在各自的存储器的空白区中存入从传输介质处接收的数据。因此可以实现一个数据处理系统，在该系统中，处理器在查找和确保一个合适的空白区的同时，以某种在线方式存储能够在传输介质上被再定位的变长度数据。

此外，各处理器具备对在存储器中所存数据再定位的功能，并在利用指示出处理器中所存数据的再定位情况的信息，通过再定位功能根据需要对数据进行再定位的同时，在各自的存储器的空白区中存贮从传输介质接收到的数据。因此，在这种情况下，在上面指出的功效之外，还能有效地利用存储器。
图 4

<table>
<thead>
<tr>
<th>识别码</th>
<th>长度</th>
<th>首地址</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16A(16B)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

图 5

<table>
<thead>
<tr>
<th>识别码</th>
<th>首地址</th>
<th>尾地址</th>
<th>空白区长短</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>低</td>
<td>低</td>
<td>d₁</td>
</tr>
<tr>
<td>i</td>
<td>fcᵢ</td>
<td>sᵢ</td>
<td>eᵢ</td>
</tr>
<tr>
<td>i+1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>n</td>
<td>-1</td>
<td>高</td>
<td>高</td>
</tr>
</tbody>
</table>
### 图 7

<table>
<thead>
<tr>
<th>识别码</th>
<th>首地址</th>
<th>层地址</th>
<th>空白区长度</th>
<th>再定位前的首地址</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-1</td>
<td>低</td>
<td>d₁</td>
<td></td>
</tr>
<tr>
<td>i</td>
<td>fcᵢ</td>
<td>sᵢ</td>
<td>eᵢ</td>
<td>dᵢ</td>
</tr>
<tr>
<td>i+1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>n</td>
<td>-1</td>
<td>高</td>
<td>高</td>
<td>-1</td>
</tr>
</tbody>
</table>

### 图 11

<table>
<thead>
<tr>
<th>识别码</th>
<th>首地址</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>n</td>
<td></td>
</tr>
</tbody>
</table>
图 10

开始

从输入数据和存储数据构成存储器分配图

如果必要，利用存储器分配图对存储数据再定位并查找和保证空白区

刷新存储器信息存储单元中的数据并存储输入数据体

将存储数据和首地址置入连接信息存储单元

从连接信息存储单元读出所需数据的首地址并访问和运行存储数据

结束