

*Ромичева Е.В., магистрант*

*Московский государственный технический университет им. Н.Э. Баумана,*

*E-mail: [helenessa96@gmail.com](mailto:helenessa96@gmail.com)*

*Ковалева Н.А., ст. преподаватель,*

*Московский государственный технический университет им. Н.Э. Баумана,*

*E-mail: [kovaleva@bmstu.ru](mailto:kovaleva@bmstu.ru)*

*Шоскальне В.М., магистрант*

*Московский государственный технический университет им. Н.Э. Баумана,*

*E-mail: [valeriya.ya.189623@gmail.com](mailto:valeriya.ya.189623@gmail.com)*

## **АВТОМАТИЗИРОВАННАЯ РАСПРЕДЕЛЕННАЯ МЕДИЦИНСКАЯ ИНФОРМАЦИОННАЯ СИСТЕМА**

**Аннотация:** В статье рассматривается возможная архитектура и структура автоматизированной распределенной медицинской информационной системы. Описан вариант реализации распределения данных между медицинскими учреждениями и обмен данными между ними. Выбраны механизмы ведения репликаций и их преимущества, а также методы управления транзакциями.

**Ключевые слова:** распределенные системы, медицинские информационные системы, СУБД, репликации, транзакции.

**Abstract:** The article discusses the possible architecture and structure of an automated distributed medical information system. A possible implementation of the distribution of data between medical institutions and the data flow mechanisms are described. The replication mechanisms and their advantages, as well as transaction management methods, were selected.

**Keywords:** distributed systems, medical information system, DBMS, replications, transactions.

Медицинские учреждения накапливают и обрабатывают огромное количество данных на бумажных носителях. Качество медицинской помощи, общий уровень жизни населения, уровень развития всей страны и каждого ее территориального субъекта зависят от эффективности использования этой информации врачами, руководителями и органами управления. При должном уровне автоматизации можно достичь высокого уровня в вопросах получения и обработки информации о пациентах и здравоохранении в целом [2; 3].

Когда пациент поступает в больницу, приходит в поликлинику или другие медицинские центры на прием к врачу или для сдачи анализов, в регистратуре проверяют, наблюдался ли пациент в этой клинике ранее. Если нет, то заводится новая карточка (а если пациент посещал множество клиник, то и не одна), если да, то врач может по предыдущим записям узнать историю болезни пациента. Но часто возникают ситуации, когда записи и даже целые карты теряются, а если пациент приходит в другую клинику, врач уже не может оценить состояние здоровья по записям, только со слов пациента. А если карта содержит много записей, то искать конкретную информацию крайне неудобно и занимает много времени, а результат поиска не всегда будет успешным.

В распределенной базе данных информация хранится в нескольких местах. Каждый такой фрагмент управляется независимо от СУБД. С помощью распределенных транзакций пользователи должны иметь возможность выполнять ежедневные запросы к базе данных, обновлять данные или выполнять прочие манипуляции с данными. Влияние транзакций на распределенные данные должно быть атомарным, то есть транзакция изменяет/получает данные успешно, если была зафиксирована, или изменения не происходят, если транзакция не была зафиксирована. Должна быть предусмотрена возможность работы в условиях интенсивного трафика, а задержки передачи данных минимизированы, чтобы удовлетворять

допустимому времени отклика, а это потребует реализации специального механизма, связанного с административными издержками и производительностью СУБД [1; 5].

Распределенная медицинская информационная система (МИС) может быть использована в одном из субъектов Российской области – регионе, области.

Нижний четвертый уровень системы представлен поликлиниками и диагностическими центрами, которые относятся к тому или иному району/городу/региону. Все поликлиники и диагностические центры обращаются к соответствующей части (фрагменту) виртуального сервера для осуществления хранения и обработки данных.

Для обеспечения надежности и отказоустойчивости, а также эффективного обращения к информации, определенные данные с серверов поликлиник реплицируются на сервер соответствующей районной больницы. Оставшаяся часть данных относится только к поликлинике и подвергается процедуре резервного копирования.

Данные о результатах анализов передаются из диагностического центра в больницу или поликлинику, которая запрашивала это лабораторное исследование, и записываются в электронную карту пациента. Таким образом осуществляется реплицирование данных о лабораторных исследованиях и обеспечивается надежность.

Третий уровень представлен районными больницам, каждая из которых также обращается к собственной части (фрагменту) виртуального сервера районных больниц. Локальность ссылок достигается за счет того, что часть данных о пациентах соответствующего района реплицируется на сервер районных больниц, поэтому при необходимости данные о пациентах можно получить напрямую без обращения к серверу поликлиник.

Аналогичным образом для обеспечения надежности, отказоустойчивости и эффективной обработки информации, определенные данные с серверов районных больниц реплицируются на сервер соответствующей городской

больницы. Оставшаяся часть данных относится только к районной больнице и подвергается процедуре резервного копирования.

Второй уровень представлен городскими больницами. Процедура репликаций и эффективного обращения к данным осуществляется аналогично районным больницам.

Верхний уровень представлен региональной больницей. На сервере больницы расположены данные, относящиеся к самой больнице, а также данные о пациентах из поликлиник соответствующего региона и городских больниц.

Все данные, относящиеся к региональной больнице, подвергаются процедурам резервного копирования на выделенный сервер, который расположен в самой больнице.

В распределенной МИС используется горизонтальная фрагментация. При таком подходе данные/записи фрагментированы по горизонтали, т.е. горизонтальное подмножество данных таблицы создается и хранится в другой базе данных в распределенной системе.

Фрагменты таблиц размещаются локально в соответствующем медицинском учреждении, при обращении к записям не будет увеличиваться нагрузка на сеть, а сами данные будут получены или изменены за минимальный промежуток времени.

Механизмы репликаций используются для обеспечения высокой доступности и производительности – нужны интегрированные системы, где репликация данных эффективно дополняется репликацией вычислений и коммуникаций.

В распределенной МИС будет использоваться репликация на основе журналов. Все проведенные транзакции (добавление/изменение записей в карте пациента, проведенные исследования и т.п.) фиксируются в журнале транзакций, а когда приходит время для синхронизации между ведущим и ведомым серверами, данные на ведомом сервере обновляются путем чтения записей из этого журнала. Данный метод эффективен тем, что в лечебном

учреждении наблюдалось или наблюдается множество пациентов, но в момент между двумя синхронизациями будут обновлены данные только о тех пациентах, лечение которых осуществлялось в этот промежуток, а старые данные о предыдущих пациентах не были обновлены.

Уровень локальных ссылок будет высок, если на узлах размещены те элементы данных, которые чаще всего используются на этих узлах. В случае аварийной ситуации на одном из узлов может произойти отказ в доступе только к той части данных, которая на нем хранилась, но работоспособность всей системы не будет нарушена. При выходе из строя ведомого узла достаточно переключить все приложение на работу с узлом-мастером до восстановления и перезапуска узла.

В качестве стратегии репликации выбран метод master-slave репликации. Изменения данных, происходящие на мастере, повторяются на репликах (но не наоборот). Поэтому запросы на изменение данных (crud-запросы) выполняются только на мастере, а запросы на чтение данных могут выполняться как на репликах, так и на мастере.

Когда пациент попадает в районную больницу, врач должен получить данные о лечении этого пациента в поликлинике, к которой он прикреплен, чтобы получить историю болезни пациента и сведения из его электронной медицинской карты. Для удобства и надежности копии данных из поликлиник сохраняются на сервере районной больницы. Таким образом обеспечивается надежность системы, и если откажет сервер поликлиники, то данные будут доступны из связанной районной больницы. Аналогично если пациент попадает в городскую больницу, данные о предыдущих обращениях пациента в медицинские учреждения дублируются на сервер городской больницы, а в случае с региональной больницей данные реплицируются с сервера городской больницы на сервер региональной больницы.

Если в ходе лечения пациента данные обновляются, то ведущий сервер обновляет данные на подчиненном сервере согласно настройкам синхронизации данных. Подчиненные узлы не меняют данные на ведущих

узлах, а ведут данные о пациенте в отдельной таблице (или фрагменте). Таким образом, имеется вся история болезни пациента, где можно отслеживать динамику изменения показателей здоровья пациента, а также определять поставленные ранее ошибочные диагнозы.

Так, сервер поликлиники выступает ведущим для сервера районной больницы, сервер районной больницы выступает ведущим для сервера городской больницы, а сервер городской больницы - для сервера региональной больницы.

К районной больнице прикрепляются все районные поликлиники. Если пациент, лечился в поликлинике, не прикрепленной к данной районной больнице, то сервер районной больницы обращается к напрямую к серверу этой поликлиники, при этом данные не реплицируются с сервера поликлиники на сервер больницы. Они доступны в режиме чтения. Врач районной больницы на основе этих данных делает соответствующие записи о состоянии и лечении пациента в БД районной больницы.

Все операции, выполняемые с данными на SQL сервере, происходят в контексте транзакций. Для отслеживания всех транзакций в распределенной СУБД используется журнал транзакций. Содержащаяся в нем информация используется для восстановления данных при прерывании выполнения операций или системном сбое (проблема сети, поломка оборудования) [4].

Если несколько пользователей одновременно осуществляет доступ (на чтение и запись) к совместно используемой базе данных, то для поддержки согласованного состояния данных требуется синхронизовать доступ. Синхронизация достигается путем применения механизма блокировок. Он гарантируют эксклюзивный доступ использования данных только в конкретной транзакции.

Уровень изолированности транзакций `read committed` (чтение фиксированных данных) обеспечивает защиту от чернового, «грязного» чтения. Этого уровня изолированности транзакций будет достаточно, поскольку к одной записи о пациенте на сервере медицинского учреждения в один момент

времени с большой вероятностью будет обращаться только 1 человек. Реплики этих данных хранятся на сервере репликаций в медицинском учреждении более высокого по иерархии уровня, эти данные обновляются с заданной периодичностью. Если вдруг к данным обратятся напрямую из другого медицинского учреждения (за неимением там записей о пациенте на сервере репликации), то блокировки чтения записи будет достаточно для предотвращения возникновения несогласованных результатов.

В распределенной МИС чаще всего информацию о пациенте запрашивает конкретный лечащий врач, поэтому редко возникают ситуации, когда к записям пытаются обратиться одновременно несколько пользователей. В случае, когда вероятность одновременного обращения несколькими пользователями к одной таблице или записи велика (например, обращение к таблице медикаменты, процедуры), используются механизмы предотвращения тупиков.

Благодаря распределенным системам можно эффективно организовать работу медицинских учреждений и сократить бумажный документооборот. Продуманная реализация подобных систем позволит решить проблему поиска данных о пациентах, их лечении и истории болезни и т.п., что позволит врачам быстро собирать историю болезни пациентов, а органы здравоохранения смогут получать статистические данные об уровне заболеваемости и медицины в данном регионе.

### **Библиографический список:**

1. Бёрнс Б. Распределенные системы. Паттерны проектирования. — СПб.: Питер, 2019. — 224 с.
2. Информационные технологии в медицине [Электронный ресурс] : —Режим доступа: <https://medbe.ru/materials/obshchestvennoe-zdorove-i-zdravookhranenie/klassifikatsiya-meditsinskikh-informatsionnykh-sistem/> (дата обращения 20.03.2020).

3. Ключ к интеграции медицинских информационных систем [Электронный ресурс]: — Режим доступа: <https://www.osp.ru/os/2014/02/13040045/> (дата обращения 20.03.2020).

4. Кузнецов С.Д. Основы баз данных. — 2-е изд. — М.: Интернет-Университет Информационных Технологий; БИНОМ. Лаборатория знаний, 2007. — 484 с.

5. Осипов Д.Л. Технологии проектирования баз данных. — М.: ДМК Пресс, 2019. — 498 с.