

информационные системы, обрабатывающие сложноструктурированные данные, и автоматизировать динамически формируемые процессы.

IBM Notes обладает встроенным механизмом репликации, что позволяет территориально удаленным подразделениям оперативно обмениваться информацией и совместно работать с документами как в интерактивном, так и в автономном режиме. Также имеется мощный механизм разделения доступа к документам, благодаря которому документ предоставляется сотруднику для просмотра или редактирования в зависимости от степени его участия в работе над документом.

IBM Notes идеально подходит для систем электронного документооборота, в особенности для территориально-распределенных организаций, за счет использования нереляционной структуры баз данных платформа реализует возможности по репликации документов между серверами и разграничению доступа к документам.

Система электронного документооборота строится из набора взаимосвязанных функциональных модулей – баз данных, предназначенных для работы с документами и выполнения служебных процедур. Каждый модуль соответствует определенному деловому процессу, связанному с обработкой информации в системе и обладает набором функций необходимых для управления соответствующим процессом. Каждому сотруднику системы предоставляется почтовый ящик и электронный адрес для использования при обмене электронной почтой со сторонними организациями, адреса сотрудников фиксируются в общедоступном справочнике организации.

**А.С. Кравченко** (УО «ГГУ им. Ф. Скорины», Гомель)

Науч. рук. **А.И. Кучеров**, старший преподаватель

## **МОНИТОРИНГ ВЫЧИСЛИТЕЛЬНОГО ПРОЦЕССА НА РАБОЧИХ СТАНЦИЯХ ЛВС**

В настоящее время каждая организация обладает внушительным парком вычислительной техники. Всё больше рабочих задач возлагаются на ЭВМ, отсюда возникает потребность в поддержании работоспособности компьютеров и сокращении времени простоя при их ремонте. В ГГУ им. Ф. Скорины насчитывается больше сотни компьютеров и уследить за всеми – очень сложная задача. Для облегчения этой задачи был разработан комплекс мониторинга вычислительного процесса на рабочих станциях. Данный комплекс существенно облегчит работу системному администратору и поможет определить время «приблизительного» износа

оборудования для своевременной замены. Устройства даже в пределах одной аудитории могут использоваться с разной интенсивностью, и именно данный комплекс поможет определить время проведения профилактических работ и риск выхода из строя компонентов рабочей станции.

Также комплекс ведёт сбор сведений о пользователе. Такая информация, как время входа и выхода, снимок активных процессов и т. д. поможет составить представление о работе данного пользователя, т. е. вывести его уникальный «почерк». В дальнейшем картина поведения пользователя может использоваться как дополнительное средство аутентификации.

Комплекс разделён на две части: клиентская и серверная.

Серверная часть посылает запрос на компьютер и получает данные об активности пользователя на данной машине (рисунок 1)

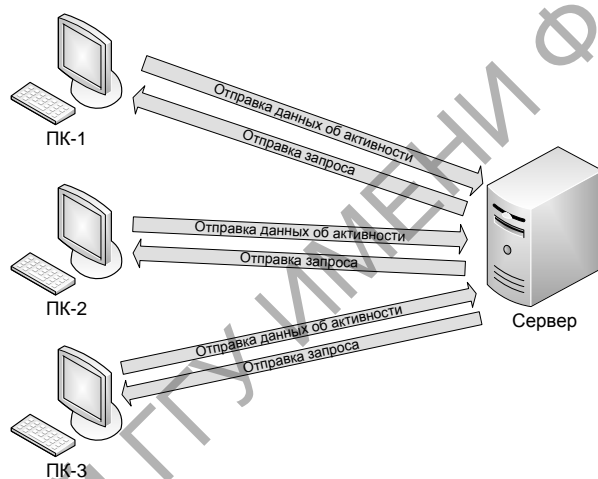


Рисунок 1 – Принцип работы комплекса

Клиентская часть представляет собой приложение win32 разработанное в visual studio на языке visual c++. Приложение выполняет задачу сбора сведений о пользователе и об использовании аппаратных средств. Так по каждому пользователю будет собрана следующая информация:

- Время входа/выхода пользователя в систему.
- Количество нажатий клавиш на клавиатуре.
- Количество нажатий клавиш мыши.
- Снимок активных процессов.
- Информация об аппаратной части компьютера.
- Информация об ОС.

Эта информация будет сохраняться в файле, который расположен в папке к которой имеет доступ сервер.

Отдельно разработана серверная часть комплекса – приложения для сбора статистики и размещения ее в базе данных. Принцип работы прост: в расшаренную папку клиентская часть слагивает логи

действий пользователя, а, собственно, данная программа эти данные собирает и распределяет по базе данных.

По таким параметрам, как количество нажатий, создано ограничение. При достижении лимита нажатий на определенном ПК системному администратору будет выведено сообщение о том, что на данном компьютере в скором времени возможно нужно будет заменить клавиатуру или мышь, т. е. необходимо иметь запасные устройства на складе. Также суммируется время работы каждого компьютера и при достижении, установленного лимита времени, будет выдано сообщение о необходимости проведения профилактических работ. В дальнейшем планируется ввод опроса ID внешнего накопителя, так за каждым пользователем будет условно закреплена его личная флешка, и в случае несовпадения ID система будет требовать подтверждения подлинности пользователя повторным вводом пароля.

В качестве СУБД была выбрана MySQL. В качестве средства доступа в БД используется ODBC драйвер. Перейти на другую СУБД труда никакого не составит, по тому, как большинство современных баз данных поддерживают этот интерфейс. Программа написана на языке Java. Во время разработки использовалась версия jdk 1.6, однако созданные классы без труда соберутся и под 1.5. Отсюда вытекают минимальные требования для гарантированной работы приложения: jdk версии 1.5 и выше. Класс драйвера для доступа к MySQL расположен рядом с проектом. Проект представляет собой несколько классов и bat-файл для его запуска.

Параметры, которые необходимо задавать программе, расположены в файле `watcher.properties`. Первый параметр задает расположение файла со списком ip-адресов машин, с которых будет информация собираться, а вторым – имя расшаренной папки.

Что касается файла с IP-адресами, то это простой текстовый файл с адресами, записанными каждый на новой строке. Программа берет список этих адресов, ищет на каждой машине соответствующую папку, анализирует содержимое этой папки. На основании этого содержимого делается вывод о том, какой пользователь работал и информацию о нем заносит в базу.

Следует отметить непригодность использования данной программы в домашних сетях. По тому как если на нескольких машинах работают пользователи с одинаковыми именами (например Admin), то информацию о них будет записываться как об одном пользователе.

Исходя из всего выше сказанного, вырисовывается логика работы всего комплекса сбора статистики: клиентская часть собирает данные, складывает их в определенную папку, программа для сбора этих данных по расписанию просматривает эти папки и заносит их в базу данных.