

Винтайкина Дарья Андреевна, студент
Самарский государственный технический университет,
г. Самара
email: total2094@mail.ru

ПРОБЛЕМЫ ЭФФЕКТИВНОСТИ ИНТЕРФЕЙСОВ ПРОГРАММНЫХ СИСТЕМ

Аннотация: Эта статья исследует проблемы эффективности интерфейсов программных систем и предлагает методы повышения производительности интерфейса на примере системы распределенного контроля за запуском приложений. В статье рассмотрены основные компоненты интерфейса и анализируются возможные узкие места в производительности. Затем предлагаются методы оптимизации интерфейса, включая улучшение работы с данными, уменьшение времени ответа и повышение удобства использования.

Ключевые слова: интерфейс, производительность, система распределенного контроля, запуск приложений, оптимизация интерфейса.

Annotation: This article explores the problems of efficiency of software system interfaces and suggests methods to improve interface performance using the example of a distributed application startup control system. The article discusses the main components of the interface and analyzes possible performance bottlenecks. Then methods for optimizing the interface are proposed, including improving data handling, reducing response time and improving usability.

Keywords: interface, performance, distributed control system, application launch, interface optimization.

Интерфейс является ключевым элементом любой программной системы и

может оказывать значительное влияние на ее производительность. Несмотря на то, что многие разработчики уделяют большое внимание проектированию интерфейса, возможны ситуации, когда производительность системы замедляется из-за неэффективного интерфейса. Эта статья представляет подробный анализ возможных проблем производительности интерфейса на примере системы распределенного контроля за запуском приложений и предлагает методы их решения.

Основными компонентами интерфейса являются элементы управления, включая кнопки, поля ввода, переключатели и списки. Кроме того, интерфейс может содержать элементы отображения, такие как таблицы, графики и диаграммы. Каждый из этих компонентов может повлиять на производительность системы.

Одним из наиболее распространенных узких мест в производительности интерфейса является работа с данными. Если приложение часто обращается к базе данных или сетевому хранилищу, это может привести к замедлению работы интерфейса. Кроме того, длительное время ответа, например, при загрузке больших файлов или при обработке сложных запросов, также может снижать производительность системы.

Существует несколько методов оптимизации интерфейса, которые могут помочь повысить его производительность. Один из таких методов - это оптимизация работы с данными. Это может включать уменьшение количества запросов к базе данных или сетевому хранилищу, а также использование кэширования данных для ускорения доступа к ним. Также можно использовать асинхронные запросы для уменьшения времени ответа и повышения отзывчивости интерфейса.

Другим методом оптимизации является уменьшение времени ответа. Это может быть достигнуто путем оптимизации кода и уменьшения количества операций, выполняемых при каждом запросе. Также можно использовать компрессию данных, чтобы уменьшить объем передаваемых данных и ускорить время ответа.

Наконец, повышение удобства использования интерфейса также может способствовать увеличению его производительности. Это может быть достигнуто путем оптимизации навигации, упрощения процессов взаимодействия с пользователем и улучшения дизайна интерфейса. В системе распределенного контроля за запуском приложений могут возникать проблемы производительности интерфейса при выполнении операций с базой данных и при передаче больших объемов данных между серверами. Для оптимизации интерфейса можно использовать следующие методы:

- Использование кэширования данных на серверах приложений для ускорения доступа к данным и уменьшения количества запросов к базе данных.
- Оптимизация кода на серверах приложений для уменьшения времени ответа и количества операций, выполняемых при каждом запросе.
- Использование асинхронных запросов для уменьшения времени ответа и увеличения отзывчивости интерфейса.
- Улучшение дизайна интерфейса, и оптимизация навигации для повышения удобства использования и сокращения времени, затрачиваемого на выполнение задач.

Давайте опираться на метод контроля, основанный на сочетании мониторинга и аудита, при котором аудит записывает различные операции любого пользователя в определенные файлы [2].

Процесс запуска и запуска программы включает в себя несколько этапов:

1. Пользователь запускает эту программу.
2. На экране появится главное окно программы. Консоль состояния системы отображает все действия, которые происходят в программе. (На рисунке 1 показано "Главное окно программы").

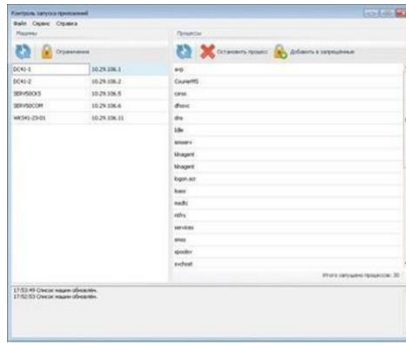


Рисунок 1- Главное окно программы

3. Чтобы остановить необходимый процесс, нажмите кнопку "Остановить процесс". Чтобы добавить процесс в список запрещенных, нажмите кнопку "Список запрещенных". После этого откроется окно "Настройки блокировки процесса".

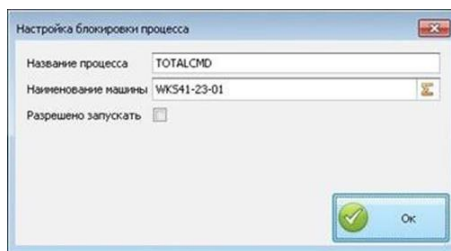


Рисунок 2 - Окно настроек блокировки процесса

4. Просмотрите ограничения выбранной машины. После этого на экране появится окно "Список настроек блокировки процесса" (меню "Список настроек блокировки процесса" показано на рисунке 3).



Рисунок 3 - Список настроек блокировки процесса

5. Чтобы удалить процесс из списка запрещенных, нажмите кнопку "Удалить". Выбранный процесс удаляется из списка запрещенных. (На рисунке

4 показано "Окно подтверждения удаления").

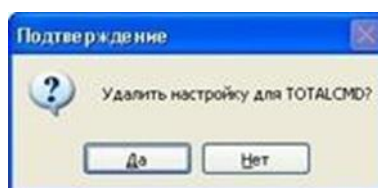


Рисунок 4 -Окно подтверждения удаления процесса из списка запрещенных

6. При нажатии кнопки "Редактировать" откроется окно "Настройки блокировки процесса", в котором настроена блокировка процесса. (Окно меню "Настройки блокировки процесса" показано на рисунке 5).

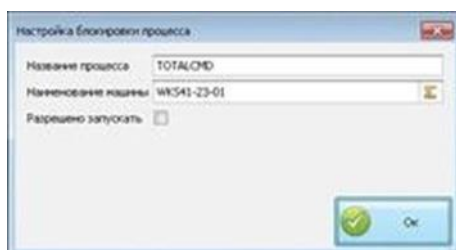


Рисунок 5 - Настройки блокировки процесса

Из-за широкого использования информационных систем программа требует много времени для работы. В связи с этим наша задача - усовершенствовать эту программу, а точнее, свести к минимуму затраченное время.

Процесс повышения эффективности пользовательского интерфейса программы будет осуществляться с использованием прототипа программного обеспечения CogTool. Программа создает простейший макет необходимого пользовательского интерфейса, а также позволяет оценить его эффективность. Программный инструмент прототипирования CogTool определяет время, затрачиваемое пользователями при выполнении определенных задач с использованием изучаемого интерфейса.

Мы загружаем интерфейс исходной программы в систему прототипов CogTool, выбираем целевую часть, делаем отображение перехода между

страницами интерфейса к нам, а также описываем модель поведения пользователя. Таким образом, мы видим, что пользователям требуется около 24,0 минут, чтобы использовать эту программу.

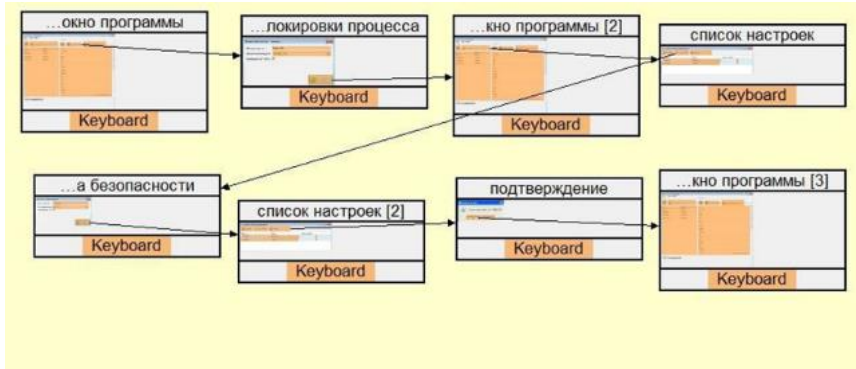


Рисунок 6

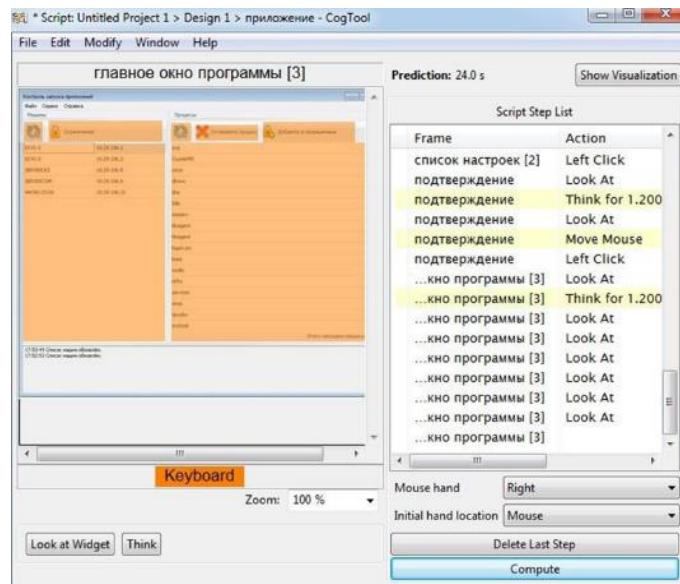


Рисунок 7-Фрагмент модели поведения пользователя при просмотре списка компьютеров и процессов

В качестве альтернативы интерфейс программы был загружен в CogTool. В результате исследования было установлено, что у пользователей это занимает 3,3 минуты.

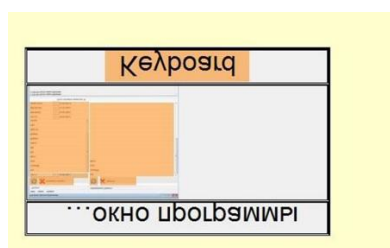


Рисунок 8

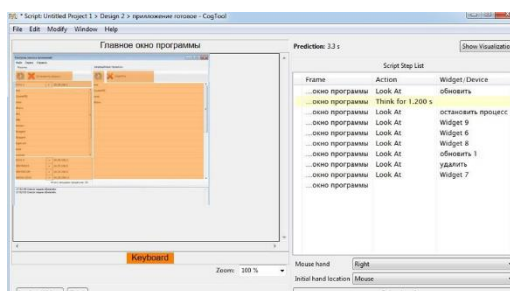


Рисунок 9-Фрагмент модели поведения пользователя при просмотре главного окна программы

Благодаря результатам нашего исследования интерфейса получается, что по сравнению с использованием оригинальной программы время, необходимое для внедрения программных средств с использованием веб-ориентированной системы, сокращается примерно в 7,5 раз.

Повышение производительности интерфейса является важной задачей для любой программной системы. В этой статье были рассмотрены основные компоненты интерфейса, возможные проблемы производительности и методы их решения на примере системы распределенного контроля за запуском приложений.

Библиографический список:

1. Справочник Adobe Photoshop – [Электронный ресурс]. – Режим доступа: <http://helpx.adobe.com/ru/photoshop/user-guide.html> (дата обращения: 12.03.2021).
2. Учебник Figma – [Электронный ресурс]. – Режим доступа: <https://figmadesign.ru/1-0-0-obzor-figma.html> (дата обращения: 12.03.2021).
3. Figma для верстальщика – [Электронный ресурс]. – Режим доступа: https://htmlacademy.ru/blog/boost/tools/figma?utm_source (дата обращения: 13.03.2021).