

## Реинжиниринг геоинформационной системы компании ООО «Владлинк»

Бессонов Илья Игоревич,  
специалист 5 курса, кафедры информационных технологий и систем  
Владивостокский государственный университет экономики и сервиса  
Россия. Владивосток  
E-mail: iliyabessonov@gmail.com; тел.: +79644343880  
ул. Гоголя, 41, г. Владивосток, Приморский край, Россия, 690014

*В статье описывается задача реинжиниринга геоинформационной системы компании ООО «Владлинк». Геоинформационная система применяется в компании для проектирования сетевой инфраструктуры и просмотра данных об оборудовании на определённом участке сети. Приведены особенности архитектуры приложения и возможные подходы для выполнения реинжиниринга.*

**Ключевые слова и словосочетания:** реинжиниринг, подходы реинжиниринга, геоинформационная система, функции ГИС, создание новой информационной системы, модификация существующей информационной системы, адаптация готовой информационной системы.

### Reengineering of geographic information system of the company «Vladlink»

Bessonov Ilya Igorevich,  
specialist of the 5<sup>th</sup> year, information technologies and systems department  
Vladivostok State University of Economics and Service  
Russia. Vladivostok

*The article describes the task of reengineering geographic information system of the company «Vladlink». Geographic information system is used in the company for the design of the network infrastructure and view data about the equipment on a certain part of the network. Described features of the application architecture, and possible approaches to implementation of reengineering.*

**Keywords:** reengineering, reengineering approaches, geographic information system, GIS functions, the creation of a new information system, modification of the existing information system, adapting the final information system.

Рассматривается задача реинжиниринга геоинформационной системы компании ООО «Владлинк». Основная деятельность компании ООО «Владлинк» заключается в предоставлении услуг Интернета и телевидения абонентам. В ходе осуществления своей деятельности, компания выполняет задачи по модернизации, ремонту и территориальному расширению сетевой инфраструктуры. Для повышения эффективности выполнения перечисленных задач отделом информационных технологий, была разработана геоинформационная система. В результате разработки была получена интерактивная визуальная модель инфраструктуры сети. К функциям ГИС можно отнести следующие:

- проектирование сетевой инфраструктуры компании;
  - просмотр данных об оборудовании на конкретных участках сети, количестве абонентов в домах и состоянии подключаемых домов к общей сети, а также выполняемых в них работах на текущий момент.
- Несмотря на широкие функциональные возможности системы, система имеет ряд недостатков:
- необходимость постоянного ручного обновления данных проектировщиком при изменениях на каких-либо участках сети;
  - отсутствие данных о выполненных действиях пользователями.
- Для устранения указанных недостатков необходимо выполнить реинжиниринг системы. При осуществлении реинжиниринга может применяться один из следующих подходов[1]:
- создание новой ИС («с чистого листа») взамен существующей;
  - модификация существующей ИС;
  - адаптация готовой ИС стороннего разработчика.

В пользу подхода разработки «с чистого листа», говорят следующие основные доводы:

- процесс создания ИС достаточно хорошо изучен, существует ряд моделей, описывающих порядок действий;
- процесс создания новой информационной системы лучше прогнозируется (по времени, стоимости и другим ресурсам), чем процесс перепроектирования;
- создание новой информационной системы позволяет отказаться от устаревших концепций и применить новые ИТ.

Однако этот подход обладает следующими существенными недостатками:

- создание новой системы требует значительных ресурсов;

- необходимо наличие резерва времени (к реинжинирингу часто прибегают в самый последний момент, когда резерва по времени почти не осталось);
- высока вероятность того, что часть задач придется решать заново;
- также высока вероятность, что часть задач в старой ИС была решена на хорошем или приемлемом уровне.

Обычно такой подход применяется в ситуации, когда по оценкам придется перепроектировать более половины компонентов информационной системы. Поэтому, большее распространение получил путь модификации существующей информационной системы.

К его основным достоинствам можно отнести:

- потенциально меньшие затраты;
- потенциально меньший срок окончания проекта;
- возможность широкого использования компонентов существующей информационной системы, функционирующих хорошо или удовлетворительно (тем самым сэкономить на разработке новых компонентов);
- минимальные затраты на переобучение пользователей;
- более плавный переход со старой информационной системы на новую (актуально для задачи переноса данных из старой системы в новую);
- возможность не останавливать информационную систему полностью (в каждый момент реинжиниринга могут быть неработоспособны отдельные подсистемы при общей работоспособности информационной системы в целом).

К существенным недостаткам этого подхода можно отнести:

- отсутствие гарантии, что путем модификации удастся привести информационную систему в соответствие к новым требованиям;
- сложность прогнозирования процесса, оценки необходимых ресурсов;
- сложность структурной модели, описывающей реинжиниринг информационной системы;
- довольно высокую вероятность возникновения «волны изменений» (модификация одного компонента требует модификации других компонентов, взаимодействующих с ним, те в свою очередь также требуют модификации связанных компонентов и таким образом количество компонентов лавинообразно растет).

К ограничениям такого подхода следует также отнести правовой аспект. Исполнителю необходимо иметь доступ к исходным текстам информационной системы, а также иметь право исследовать и изменять информационную систему.

В последнее время, с появлением открытых (свободно распространяемых) программ и информационных систем, распространение получил аналогичный подход – путем «адаптации» (доработки) готовой ИС стороннего разработчика. Обычно такие информационные системы строятся по технологии так называемых «открытых систем», что существенно упрощает модификацию системы.

Этот подход можно рекомендовать в следующих случаях:

- наличие аналогичной информационной системы, требующей минимальной доработки;
- ожидание существенных выгод от использования существующей системы (например, получение совместимости с другими важными информационными системами);

- наличие в существующей информационной системе хорошего набора базовых функций (так называемого «ядра»), на основе которого упрощается реализация новой информационной системы.

Для выполнения поставленной задачи наиболее оптимальным является подход модификации существующей информационной системы, так как он удовлетворяет следующим критериям:

- ограниченность срока на выполнение реинжиниринга;
- отсутствие необходимости остановки существующей системы, в период осуществления реинжиниринга.

Помимо этого, добавляемый функционал будет являться расширением функционала существующей системы, то есть большая часть компонентов может быть заимствована из действующей системы.

Данная ГИС система представляет собой клиент-серверное приложение. Функционал клиентской части выполняют html-страницы и js-скрипты, полученные после ретрансляции GWT- компилятором. А функционал серверной части является объединением трех различных подсистем реализованных на Java, Perl, Python языках.

Подсистема «Вывод детализированной информации», написанная на языке Python, создана на основе веб-фреймворка Django. Она отвечает за вывод всплывающих окон с детализированной информацией об объектах, при наведении курсора мыши на них.

Подсистема «Операции с объектами на карте», написанная на языке Perl, реализует серверный функционал, связанный с выборкой, созданием, редактированием и удалением объектов на карте.

Подсистема «Основные функции ГИС», написанная на языке Java, отвечает за оставшийся

функционал:

- выборка адресов улиц и номеров домов;
- загрузка требуемых областей карты;
- создание и удаление слоёв с объектами;
- создание, редактирование, удаление муфт, кроссов и соединений волокон в них;
- поиск объектов по адресу привязки.

Добавление новых функций к ГИС системе является сложным процессом, если эти функции связаны с несколькими серверными подсистемами одновременно. В случае внесения изменений в подсистему, написанную на Java, потребуется перекомпиляция GWT-проекта целиком.

Помимо этого, необходимо поддерживать актуальность версий используемых технологических решений, на что в данном случае затрачивается много времени, так как серверные подсистемы созданы на основе различных технологий.

Таким образом, для того чтобы исправить имеющиеся технологические недостатки, перед добавлением нового функционала, необходимо реализовать переход серверных подсистем к единой технологии.

Перенос каждой из подсистем имеет свои технологические особенности. В случае с подсистемой «Вывод детализированной информации», разработанной на платформе Django, все обрабатываемые запросы и используемые модели могут быть получены из «Менеджера URL»[3]. После выполнения анализа запросов и моделей, осуществляется перенос подсистемы на требуемую платформу.

Подсистема «Операции с объектами на карте» реализована на основе технологии CGI (Common Gateway Interface) с использованием библиотеки «CGI::Simple». Данная подсистема обрабатывает четыре типа запросов - get, post, put, delete, и для каждого типа вызывается различная функция из пакета «WFS». Для каждого типа запросов также выполняется анализ исходного кода используемых функций с последующей переработкой под требуемую платформу.

Подсистема «Основные функции ГИС» имеет модульную структуру и разработана с использованием GWT-технологии. Для выполнения переноса этой подсистемы кроме анализа исходного кода серверных модулей необходимо проанализировать код клиентских модулей.

Взаимодействие клиентских и серверных модулей реализовано посредством «Request Factory» и «AsyncCallback» - библиотек поставляемых вместе с фреймворком GWT[2].

Библиотека «AsyncCallback» представляет собой механизм удалённого вызова процедур и использует подстроку «app», заданную по умолчанию, в качестве шаблона для формирования строки http-запроса. Серверный модуль, из которого вызывается функция, определяется в клиентском модуле заданием параметра «RemoteServiceRelativePath()». Строка запроса дополняется параметром «RemoteServiceRelativePath()» с названием вызываемого метода и передаваемыми аргументами. Затем выполняется сам запрос с возвращением его результата.

Библиотека «RequestFactory» является альтернативой «AsyncCallback», позволяет разрабатывать код приложения ориентированно на данные и этим обеспечить более высокий уровень абстракции. Для формирования шаблона строки запроса используется подстрока «grc», определённая в модуле «AppRequestTransport.java». Строка запроса дополняется названием клиентского модуля с названием вызываемого метода и передаваемыми аргументами, затем выполняется запрос.

После выполнения анализа соответствующих клиентских и серверных модулей, функции серверных модулей перерабатываются под требуемую платформу.

Задача реинжиниринга будет заключаться в выполнении перехода серверной части приложения к единой технологии для исправления имеющихся технологических недостатков, и последующего добавления к системе необходимого функционала.

---

1. Ахтырченко К.В, Методы и технологии реинжиниринга ИС [Электронный ресурс]. – Режим доступа: <http://www.citforum.ru/SE/project/isr>.

2. Документация разработчика по технологии Google Web Toolkit [Электронный ресурс]. – Режим доступа: <http://www.gwtproject.org/doc/latest/tutorial/buildui.html>.

3. Документация Django [Электронный ресурс]. – Режим доступа: <http://djbook.ru/rel1.4/topics/http/urls.html>.