

УДК 629.58:681.5

ПРОГРАММНАЯ ПЛАТФОРМА ДЛЯ ПОСТРОЕНИЯ СОБЫТИЙНО-ОРИЕНТИРОВАННОЙ СИСТЕМЫ УПРАВЛЕНИЯ АВТОНОМНОГО НЕОБИТАЕМОГО ПОДВОДНОГО АППАРАТА

А.Н. Севрюк, Р.А. Сенин

Институт проблем морских технологий
ДВО РАН, Владивосток

Рассматриваются вопросы организации событийно-ориентированной системы управления (СУ) АНПА на основе СУБД РВ. Использование БД позволяет упростить процесс создания, отладки и модернизации программного обеспечения АНПА. Обеспечивается также поддержка сменного оборудования АНПА, оптимизация использования ресурсов бортовой сети АНПА и облегчается интеграция СУ АНПА со средой визуального моделирования. Описываются требования к разрабатываемой СУ и механизмы их реализации, приводится пример использования унифицированного функционального интерфейса.

ВВЕДЕНИЕ

Комплексность и разнородность решаемых АНПА исследовательских задач [1] требует от разработчиков создания более совершенных методов управления, объединяющих различные способы получения и обработки информации. Система управления (СУ) при этом должна обеспечивать эффективные средства интегрирования алгоритмов управления в свой состав. От того, как реализованы эти механизмы, во многом зависят удобство эксплуатации и надежность функционирования АНПА.

В настоящее время системы управления АНПА, разрабатываемые в ИПМТ ДВО РАН, проектируются на основе трехуровневой архитектуры [1], основанной на методах функционального проектирования. Взаимодействие всей совокупности управляющих программ (драйверов) осуществляется как с использованием механизма прямого обмена информационными сообщениями, так и

области общей памяти, предназначенной для обмена данными. Оба этих механизма обеспечивают простоту и ясность информационного обмена, сопряженную с малыми требованиями к вычислительным ресурсам процессоров. Однако такой подход обладает некоторыми недостатками, к которым можно отнести «волновой эффект», отсутствие ретроспективного доступа к накопленным данным и некоторые другие [2].

Одним из способов решения указанных проблем может быть использование СУБД РВ в качестве дополнительного программного уровня. Введение централизованной базы данных позволяет не только эффективно упорядочивать информационные потоки, но и обеспечивать ретроспективный доступ к данным и механизм представления событий [2]. При этом к работе СУБД предъявляется ряд требований, которым должна удовлетворять БД для возможности её использования в составе СУ АНПА [3].

Ещё одним фактором эффективной работы СУ является простота и единообразие процессов регистрации и извлечения необходимых данных. Одним из способов достижения этой цели является введение архитектурно-независимых функций, обеспечивающих манипуляции с данными независимо от их логической структуры (сохранение структурной целостности СУ АНПА). Множество таких функций называется *унифицированным функциональным интерфейсом на основе СУБД РВ (УФИ)* [4].

В настоящее время в ИПМТ ДВО РАН активно ведутся работы по интеграции СУБД реального времени (РВ) в состав СУ АНПА. В статье рассматриваются некоторые вопросы организации программной платформы на основе СУБД для создания событийно-ориентированной системы управления АНПА, а так же ее основные отличия от СУ, используемой в данный момент.

■ ОСНОВНЫЕ ТРЕБОВАНИЯ К ПРОГРАММНОЙ ПЛАТФОРМЕ

Программная платформа (ПП) системы управления создается для обеспечения средств интегрирования управляющих и исполнительных программных модулей в состав СУ АНПА. Для выполнения этой задачи ПП должна предоставлять средства:

- взаимодействия между программными модулями СУ и ПП;
- управления данными.

Программные модули СУ должны иметь возможность подключаться к ПП и отключаться от нее в любой момент работы СУ АНПА. Взаимодействие между модулями системы и ПП должно быть реализовано с помощью УФИ. При этом интерфейс должен:

- изолировать процессы, происходящие внутри ПП, от внешних модулей;
- обеспечивать независимость внешних модулей СУ от способов представления данных внутри ПП (сохранение структурной целостности СУ [4]);
- реализовывать механизм событий [2, 3, 4]. Под событием понимается асинхронное сообщение, несущее определенную информацию о произошедших в системе изменениях и предполагающее соответствующую реакцию;
- выполнять операции извлечения, добавления и поиска данных с использованием одного и того же множества функций безотносительно к их внутренней организации.

Управление данными в ПП реализуется с использованием СУБД, обеспечивающей [2, 3]:

- производительность, достаточную для функционирования СУ АНПА;

- отказоустойчивость (сохранение целостности накопленных данных в случае программно-аппаратных сбоев в процессорной системе АНПА).

Используемая в БД модель данных должна быть адаптивной [2, 3], т.е. иметь возможность количественного и качественного изменения классов данных без изменения общей структуры модели данных. Также модель данных должна обеспечивать:

- однозначную связь параметров со временем их получения;
- произвольный доступ к значениям параметров (мгновенным и ретроспективным);
- поиск значения параметра по времени и по другому параметру;
- соответствие следующим требованиям нормализованности:
 - минимальная избыточность (состояние совокупности таблиц, при котором дублирование данных в полях по возможности отсутствует);
 - отсутствие аномалий включения (добавление нового параметра, а также изменение его признаков не потребует изменений числа полей более чем в одной таблице).

■ УНИФИЦИРОВАННЫЙ ФУНКЦИОНАЛЬНЫЙ ИНТЕРФЕЙС (УФИ)

Место УФИ в архитектуре СУ АНПА

Как было сказано выше, **УФИ** – это унифицированный интерфейс программирования робота, облегчающий конфигурирование СУ для последовательного ряда моделей АНПА и адаптивный к изменениям

конфигурации сменного оборудования. Под адаптивностью в данном случае понимается набор свойств, которые позволят сделать процесс программирования робота более простым и удобным. УФИ – это набор функций доступа к данным, обеспечивающий выполнение моделью данных указанных выше требований. Задача интерфейса состоит в изоляции модели данных от внешней среды программирования. Вне зависимости от организации данных в СУ АНПА операции извлечения, добавления и поиска данных выполняются одним и тем же множеством функций безотносительно к их внутренней организации.

Данная особенность позволяет УФИ быть основным инструментом интеграции СУ АНПА и систем моделирования, включающим подсистему моделирования сетевого оборудования аппарата (сети бортовых микроконтроллеров) [5]. Реализация алгоритмов управления робота как в среде моделирующего комплекса, так и в среде управления реального АНПА обретает общие синтаксические и семантические характеристики. Последнее обстоятельство позволяет разрабатывать и отлаживать алгоритмы управления в среде симуляции, а эксплуатировать их на АНПА. Такой подход дает возможность снизить издержки на разработку, адаптацию и поддержку программного обеспечения АНПА.

УФИ – это своего рода «чёрный ящик», входные и выходные данные которого известны, а сам механизм преобразования скрыт. Внутренняя структура функций тесно связана со структурами данных концептуальной модели [2]. На данном этапе разработки УФИ реали-

зуется в виде API-функций, используемых приложениями-клиентами (например, регуляторами, драйверами устройств и т.д.) и являющихся потребителем и/или источниками данных. Приложение-клиент использует заголовочный файл с API-функциями, через которые получает доступ к УФИ.

Применяемые в настоящее время функции УФИ можно разбить на несколько классов, перечисленных ниже.

- Инициализация компонента-источника/потребителя данных. Во время инициализации компонент системы заявляет, какими данными он пользуется по чтению и по записи. Данная спецификация не должна меняться в процессе работы.
- Подписка на события обновления данных. Конфигурация подписки на события может меняться (динамическая подписка и отписка).
- Публикация данных в системе (сохранение данных в накопителе).
- Получение данных (запрос наиболее свежих данных, либо ретроспективных данных из накопителя).

На рис. 1 работа СУ АНПА представлена в виде диаграммы вариантов использования (UML/Use Case). На диаграмме в качестве субъектов деятельности представлены приложения-клиенты, УФИ и СУБД, а объектами их деятельности являются непосредственно API-функции.

Механизм работы СУ АНПА следующий.

- Приложение-клиент получает данные (от приписанного к нему измерительного устройства и/или на основании собственных вычислений), после чего на основании заложенного в него алгоритма принимается решение о том, следует эти дан-

ные сохранить или/и отослать другому клиенту с соответствующей отметкой о характере изменения параметра (события). Клиент может начинать свою работу и завершать до начала миссии, например, в том случае, когда необходимо выполнить инициализацию БД после сбоя или перед первым запуском. Приложения-клиенты могут быть двух видов:

1. Компонент-источник данных (например, драйвер устройства, выполняющий первичную обработку данных);

2. Компонент-потребитель данных (например, управляющий алгоритм).

- СУБД упорядочивает, извлекает и регистрирует данные о параметрах в соответствии с выбранной DDL-схемой [3], исходя из требований запросов, которые получает от клиента через УФИ.

- УФИ представлен на диаграмме как абстрактный класс, т.к. единственным его действием является вызов функции, необходимой клиенту для реги-

страции или извлечения определенных данных. Функции УФИ могут быть как сложносоставными (состоять из нескольких функций), так и атомарными (см. «Инициализация БД» на рис. 1).

Организация УФИ

УФИ в качестве своей основы использует язык С. Это обстоятельство позволяет интегрировать данный интерфейс в архитектуру СУ АНПА, эксплуатируемую в настоящий момент.

Для упрощения синтаксиса и семантики функций, входящих в УФИ, в системе используются несколько структурных типов данных. В качестве последних выступают объединенные по смыслу количественные характеристики СУ АНПА (например, сведения о системе, параметре или группе параметров). Тип данных, хранящий основные сведения о СУ, содержит следующие поля:

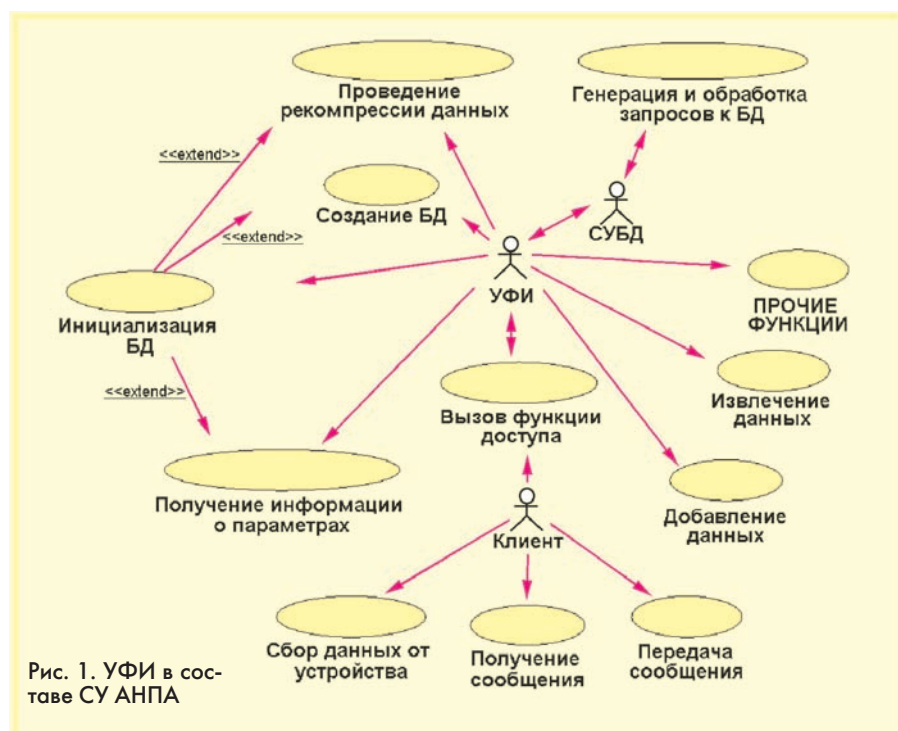


Рис. 1. УФИ в составе СУ АНПА

- число параметров в системе;
- имя БД (директории).

Приведём пример приложения, иллюстрирующего работу УФИ на прикладном уровне. Допустим, цель действий клиента – получать от измерительного устройства с определенным идентификатором набор значений параметров, заданных списком названий, и сохранять эти значения в БД. Изобразим процесс схематически, используя диаграмму IDEF0 (методология SADT [7]) низкой степени детализации, представленную на рис. 2.

На момент запуска клиент обновляет данные о последних значениях параметров при помощи функции УФИ (блок №1), используя аргументы, указанные над стрелками, входящими в блок слева. Функция возвращает результат в виде специального структурного типа (стрелка справа). Далее начинает действовать процесс получения мгновенных значений параметров от устройства (блок №2). Кратность опроса устройства регламентируется правилами получения значений (стрелка сверху). После каждого опроса устройства данные о предыдущих и мгновенных значениях параметров поступают в функцию реги-

страции в БД (блок №3). Каждое новое (текущее) значение добавляется в БД с учетом сведений о предыдущем. Это делается с целью устранения избыточности данных, получаемых от измерительных устройств в соответствии с требованиями к модели данных [3]. Избыточность возникает из-за различных по величине периодов обновления и изменения параметров. Исходя из этого параметры СУ АНПА можно разделить на группы с высокой и низкой степенью изменчивости. К первой группе относятся параметры, значения которых практически не повторяются (периоды измерения и обновления в равной степени велики). В параметрах второй группы, напротив, повторяющихся значений очень много. Исходя из принадлежности параметра к той или иной группе функция регистрации сохраняет его в БД. Упрощенно смысл устранения избыточности состоит в следующем: «Если предыдущие и мгновенные значения параметров не совпадают, то мгновенные значения добавляются в БД и предыдущие значения заменяются на мгновенные» (см. блок №3 на рис. 2). Подробнее механизм устранения

избыточности данных изложен в [3].

Одно из основных требований к УФИ говорит о том, что связь между УФИ и другими модулями СУ АНПА должна осуществляться только по данным, исключая обращения процессов напрямую к СУБД. Приведенная в примере функция (блок №3) полностью удовлетворяет данному требованию. Кроме массивов данных она содержит параметры, общие для всех СУБД. Влияние изменений внутреннего механизма доступа на формальную структуру функций исключено по той же причине. Более подробное описание требований к СУБД, подходящих для работы в СУ АНПА, можно найти в [2].

Данный подход позволяет избавиться от «волнового эффекта» в системе. Все данные о параметрах СУ АНПА хранятся в конфигурационных файлах. Изменяя аппаратную конфигурацию АНПА, можно загрузить соответствующий конфигурационный файл. На основе этих данных программная платформа автоматически создаст необходимую для сохранения информации структуру БД и определит работу приложений-клиентов.

Механизмы передачи данных в ПП

Обмен между клиентами СУ осуществляется посредством сообщений, т.е. особым образом структурированных блоков информации, принимаемых и отсылаемых процессами с помощью механизма IPC операционной системы QNX.

Программная платформа функционирует на основе ме-

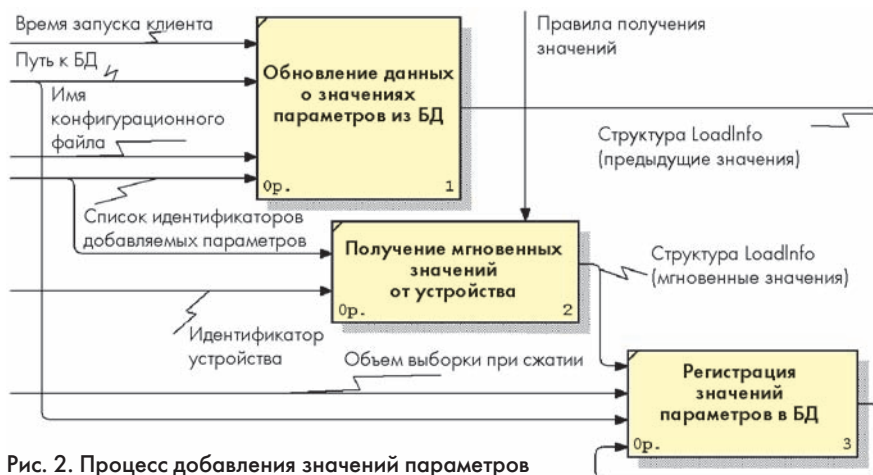


Рис. 2. Процесс добавления значений параметров

ханизма событий. Событие отсылается по инициативе клиента-источника всем подписанным на него клиентам-потребителям.

Ниже представлены структуры сообщений, которые делаются на следующие типы.

Сообщения с данными (сообщения событий) (рис. 3) представляют собой одну или несколько структур вида:

- идентификатор параметра;
- временная метка;



Рис. 3. Схема передачи сообщения с данными

- значение в виде буфера (интерпретируется общей библиотекой по идентификатору параметра).

Сообщения команд (рис. 4) имеют структуру вида:

- идентификатор команды;
- параметры команды в общем виде (интерпретируются общей библиотекой по идентификатору команды).



Рис. 4. Схема передачи команды

Команда – это сообщение, которое отсылается управляемому клиенту по инициативе управляющего клиента. Несет информацию о действиях, которые надо выполнить (код операции) и параметры этих действий (данные).

Сообщения-запросы (рис. 5)

имеют структуру:

- идентификатор запроса;
- идентификаторы параметров.

Ответ имеет структуру сообщения события.



Рис. 5. Схема работы запроса

Все сообщения могут посылаться как в синхронном, так и в асинхронном режимах.

ЗАКЛЮЧЕНИЕ

В настоящее время ведётся разработка системы управления АНПА, использующей программную платформу на основе УФИ. Заложенные возможности позволяют создать механизмы доступа к данным, обеспечивающие единообразный способ разработки миссий для аппаратов, имеющих разную конфигурацию оборудования. Используемая структура хранения данных обеспечивает эффективное использование ресурсов СУБД.

Говоря о перспективах развития данной архитектуры, необходимо отметить такую важную задачу, как интеграция с моделирующим комплексом для разработки и отладки программного обеспечения АНПА [6]. Необходимо обеспечить взаимосвязь между системой управления АНПА (или её копией) и моделирующим комплексом для отладки алгоритмов работы АНПА в среде моделирования.

Работа выполнена при финансовой поддержке грантов конкурса ДВО РАН, проекты № 06-11-04-03-002, 06-111-А-01-010.

ЛИТЕРАТУРА

1. Автономные подводные роботы: системы и технологии / М.Д. Агеев, Л.В. Киселев, Ю.В. Матвиенко и др.; под общ. ред. акад. М.Д. Агеева. М.: Наука, 2005. 398 с.
2. Инзарцев А.В., Севрюк А.Н. Использование СУБД реального времени в системе управления АНПА // Матер. 1-й междунар. науч.-техн. конф. «Технические проблемы освоения Мирового океана». Владивосток, 2005. С. 27-33.
3. Инзарцев А.В., Севрюк А.Н. Механизмы повышения эффективности функционирования СУБД реального времени в системе управления АНПА // Матер. 2-й междунар. науч.-техн. конф. «Технические проблемы освоения Мирового океана». Владивосток, 2007. С. 77-82.
4. Севрюк А.Н., Сенин Р.А. Организация информационного взаимодействия между программными модулями АНПА с использованием унифицированного функционального интерфейса на основе реляционной СУБД // Там же. С. 83-90.
5. Инзарцев А.В., Львов О.Ю., Сидоренко А.В., Хмельков Д.Б. Архитектурные конфигурации систем управления АНПА // Подводные исследования и робототехника. 2006. №1. С.18-30.
6. Инзарцев А.В., Киселев Л.В., Медведев А.В., Павин А.М., Севрюк А.Н., Бобков В.А., Борисов Ю.С., Мельман С.В. Разработка имитационного моделирующего комплекса для решения задач «интеллектуального» управления АНПА. Материалы междунар. науч.-техн. конф. «Технические проблемы освоения мирового океана», Владивосток: Дальнаука, 2007, с. 70-76, ISBN 978-5-8044-0794-1.
7. www.idef.com.