

ССУДА-Специализированная Система Управления распределенными ДАнными реального времени

В.П.Воеводин

ГНЦ РФ Институт физики высоких энергий, Протвино, Россия

Введение

Системы управления ускорителями (СУ) должны удовлетворять ряду требований, характерных как для любой системы реального времени (реагируемость, одновременность, предсказуемость и доверяемость), так и для АСУ ТП (цикличность):

1. Реагируемость — СУ гарантирует выполнение определенных действий за требуемое время.

2. Одновременность — более чем одно событие может произойти одновременно, но условие реагируемости будет удовлетворено.

3. Предсказуемость — СУ реагирует на все возможные события и их комбинации определенным предсказуемым образом.

4. Доверяемость — необходимо, чтобы все, зависящие от СУ, могли положиться на нее в своей деятельности.

5. Цикличность — процесс производства пучков заряженных частиц носит циклический повторяющийся характер.

6. Интерактивность — существенное требование для непромышленных установок, которые сами являются объектом исследования.

Из условий цикличности и предсказуемости следует, что можно заранее определить число объектов и структур данных внутри СУ, необходимых для функционирования установки и решения всех задач операторов и физиков ускорителей во время сеанса. Условие реагируемости предполагает наличие быстрого доступа к информации. Исходя из этого, в ИФВЭ были разработаны специальные средства доступа и хранения информации в виде трехмерных таблиц [1,2]. С использованием этих разработок в рамках проекта новой СУ комплекса У-70 была создана СУБД ССУДА.

1. Функциональные возможности СУБД ССУДА

ССУДА является распределенной как по хранению данных, так и по доступу к ним. На каждой ЭВМ верхнего уровня стандартной трехуровневой архитектуры СУ может быть расположено любое количество ССУДА-файлов (БД — баз данных). Файл может содержать любое количество ССУДА-таблиц. Таблица при создании определяется как локальная или глобальная. Данные локальной таблицы доступны только для программ, выполняющихся на этой же ЭВМ. Локальная таблица на данной ЭВМ однозначно определяется номером БД и именем таблицы, т.е. в разных ССУДА-файлах могут быть локальные таблицы с одинаковыми именами. Глобальная таблица доступна с любой ЭВМ и определяется своим уникальным в СУ именем.

Кроме того, создавая описание глобальной таблицы можно определить, что при ее открытии необходимо создавать копии в оперативной памяти ЭВМ среднего и/или нижнего уровней. При включении контроллера оборудования все относящиеся к нему таблицы открываются автоматически. При создании копий данные таблицы и ее сокращенное описание

копируются из ССУДА-файла в память. В дальнейшем прикладные программы (ПП) могут использовать все копии одной глобальной таблицы как различные таблицы одного формата, расположенные на разных уровнях СУ. В настоящее время практически используются только копии на ЭВМ нижнего уровня (в памяти контроллеров оборудования). ССУДА реализована на ЭВМ со следующими операционными системами:

- верхний уровень — DECstation с Ultrix, DEC Alpha с Linux и DEC Unix, IBM PC с Linux;
- средний уровень — VME каркас с процессорами Motorola и системой LynxOS;
- нижний уровень — процессоры Intel 8086/80186 с монитором реального времени VPV.

Прикладным программам, выполняющимся в контроллере оборудования, доступны только копии таблиц в памяти этого контроллера. ПП на верхнем и среднем уровнях доступны все копии всех глобальных таблиц всех ЭВМ СУ. При этом прикладные программы не знают о реальном местонахождении ССУДА-таблиц, что позволяет осуществлять перераспределение данных внутри СУ прозрачным для всех образом. Также и выполнение любой программы можно осуществлять в разных ЭВМ, что позволяет эффективно использовать вычислительные ресурсы посредством перераспределения задач по машинам СУ.

Адресация данных в любой таблице осуществляется указанием номеров начальных и конечных координат по трем измерениям — строки, столбцы, плоскости. Это позволяет осуществлять циклическую обработку данных таблицы, выполнять выборки сложной конфигурации. На каждой ЭВМ, где есть ССУДА-файлы, загружается программа-сервер, обслуживающая запросы к ССУДА-данным, хранящимся на этой ЭВМ. Данные хранятся и передаются в двоичном виде. Преобразования форматов данных для различных типов процессоров осуществляются средствами ССУДА. Протоколы доступа к распределенным данным реализованы на сетевом протоколе UDP.

Одна из ЭВМ выполняет роль сервера СУБД ССУДА. На ней находится специальная системная БД и выполняется ряд программ-серверов. Системная БД содержит описания ЭВМ, типов процессоров, местонахождение баз данных и глобальных таблиц и другую конфигурационную информацию. Программы-серверы обслуживают запросы на открытие таблиц, инициализируют создание копий таблиц в памяти ЭВМ среднего и нижнего уровней СУ, ведут журнал текущего состояния ССУДА и т.д. Одна из директорий файловой системы тех ЭВМ, на которых могут выполняться программы, осуществляющие доступ к базам данных ССУДА, должна быть смонтирована через NFS на соответствующую директорию ССУДА сервера.

ССУДА обеспечивает три степени защиты данных:

ЭВМ — доступ к данным можно осуществить только с тех ЭВМ, которые описаны в системной БД и смонтированы на сервер.

Владелец — при создании таблицы можно указать владельца, тогда запись в нее может осуществлять только программа, указавшая пароль владельца или менеджера ССУДА.

Таблица — чтобы работать с данными, программа должна знать имена соответствующих таблиц, узнать которые можно только у менеджера ССУДА.

2. ССУДА и архитектура ПО СУ У-70

Вышеуказанные возможности СУБД ССУДА позволили ей занять центральное место в архитектуре программного обеспечения (ПО) СУ комплекса У-70. Для проектирования ПО были определены следующие базовые правила:

1. Основная функция ПП — реализация алгоритмов обработки данных.

2. Отделить данные от ПП.
3. Отделить доступ к оборудованию от алгоритмов обработки.
4. Отделить пользовательский интерфейс от ПП.
5. Осуществлять межпрограммные взаимодействия исключительно через БД ССУДА.
6. Максимизация количества data driven программ.

В результате выполнения перечисленных правил ПО комплекса У-70 состоит из трех слабосвязанных (через таблицы данных) частей, которые могут развиваться независимо друг от друга:

- а) алгоритмы обработки и управления работают только с двумя типами формальных абстрактных объектов — вектор и структура;
- б) интерфейс с пользователем един для всего ускорительного комплекса, реализован в виде отдельной программы;
- в) доступ к оборудованию реализован исключительно внутри контроллеров оборудования, его основная функция — это обмен данными между аппаратурой и таблицами ССУДА в памяти контроллера.

Все взаимодействия осуществляются через чтение/запись определенных структур и векторов в таблицах ССУДА. Развитие наиболее трудоемкой и масштабной части (а) осуществляется посредством создания новых абстрактных объектов внутри СУ и написанием соответствующих программных компонент без внесения каких-либо изменений в другие части ПО. Она в значительной степени определяет, в какой степени СУ отвечает требованиям предсказуемости и доверяемости. Выработанные правила позволяют принципиально изменить организацию прикладного программирования.

Традиционная прикладная программа содержит в себе все три части: интерфейс с человеком, обработку данных, доступ к оборудованию. В нашей схеме прикладные программисты верхнего и среднего уровней СУ должны реализовывать только алгоритмы контроля и управления. При этом любой алгоритм можно реализовать как отдельный программный компонент, реализующий операцию над множеством объектов ССУДА, указанных внешними параметрами. Тогда решение каждой задачи оператора или физика ускорителя можно представить как рабочее обрамление (framework), строящееся на базе существующих компонент с применением некоторых технологий одного из направлений объектно-ориентированного проектирования [3,4]. Появление новой, более совершенной аппаратуры, которая в определенной степени влияет на реагируемость СУ, вызовет изменение только в соответствующих программах контроллеров оборудования. Пользовательский интерфейс, определяющий степень интерактивности СУ, может развиваться заменой внутри всего лишь одного framework, реализующего пользовательский интерфейс, некоторых компонент, которые будут построены на базе более мощного графического инструментария.

Таким образом СУБД ССУДА создает интегрированную вычислительную среду с единым интерфейсом взаимодействия всех программных компонент на всех уровнях СУ.

Список литературы

- [1] Воеводин В.П., Детиненко А.Ф., Ковальцов В.И., Пузынин В.И. — Препринт ИФВЭ 88-109, Серпухов, 1988.
- [2] Воеводин В.П., Детиненко А.Ф., Ковальцов В.И., Пузынин В.И. — М.: Наука, Программирование, №5, 1989, с.91-95.
- [3] Gamma E., Helm R., Johnson R., Vlissides J. Design Patterns: Elements of Reusable Object-Oriented Software. — Addison Wesley, October 1994.
- [4] Buschmann F., Meunier R., Rohnert H., Sommerlan P., Stal M. Pattern-Oriented Software Architecture-A system of Patterns. — Wiley, 1996.