

DOI: <https://doi.org/10.30898/1684-1719.2022.7.1>

УДК: 004.057

СОВМЕСТИМОСТЬ И ПЕРЕНОСИМОСТЬ ДАННЫХ ПРИ ОБЕСПЕЧЕНИИ ТЕХНИЧЕСКОЙ ИНТЕРОПЕРАБЕЛЬНОСТИ СЕТЕЦЕНТРИЧЕСКИХ СИСТЕМ

С.И. Макаренко

Санкт-Петербургский Федеральный исследовательский центр РАН
199178, Россия, Санкт-Петербург, 14 линия В.О., д. 39

Статья поступила в редакцию 17 мая 2022 г.

Аннотация. В условиях перехода информационно-управляющих систем к сетецентрической архитектуре и созданию сетецентрических информационно-управляющих систем (СЦИУС) возрастает актуальность обеспечения интероперабельности в таких системах. В статье проведено исследование обеспечения совместимости и переносимости данных при реализации технической интероперабельности СЦИУС с учетом жизненного цикла данных. Представлены рекомендации по обеспечению совместимости и переносимости данных на этапах формирования, передачи, хранения, обработки и представления данных в СЦИУС.

Ключевые слова: интероперабельность, сетецентрическая система, совместимость данных, переносимость данных, жизненный цикл данных, формирование данных, передача данных, хранение данных, обработка данных, представление информации.

Финансирование: Результаты, представленные в данной работе, получены в рамках госбюджетной темы НИР FFZF-2022-0004.

Автор для переписки: Макаренко Сергей Иванович, mak-serg@yandex.ru

Введение

В настоящее время развитие информационных систем ведется в направлении их интеграции и глобализации. Особенностью развития систем управления организационными и техническими системами является переход их к сетевидческому принципу построения. Системы управления, построенные в соответствии с сетевидческим принципом, получили наименование сетевидческих информационно-управляющих систем (СЦИУС).

При этом основным принципом построения СЦИУС служит интероперабельность – способность двух или более информационных систем или компонентов к обмену информацией и к использованию информации, полученной в результате обмена [1, 2].

Следует отметить, что несмотря на то, что в отечественной литературе достаточно большое число публикаций посвящено обсуждению принципов формирования СЦИУС, в основном в военной области, в то же время работ по проблеме интероперабельности в СЦИУС несравненно меньше. К основным из них следует отнести работы [3-14], при этом до настоящего времени достаточно глубокое исследование влияния аспектов совместимости и переносимости данных при оценке технической интероперабельности СЦИУС в отечественной науке не проводилась. В настоящей статье сделана попытка проведения такого исследования.

В работе [5] обоснован общий вариант декомпозиции параметров интероперабельности СЦИУС в соответствии ГОСТ Р 55062-2012 [2] – см. рис. 1.

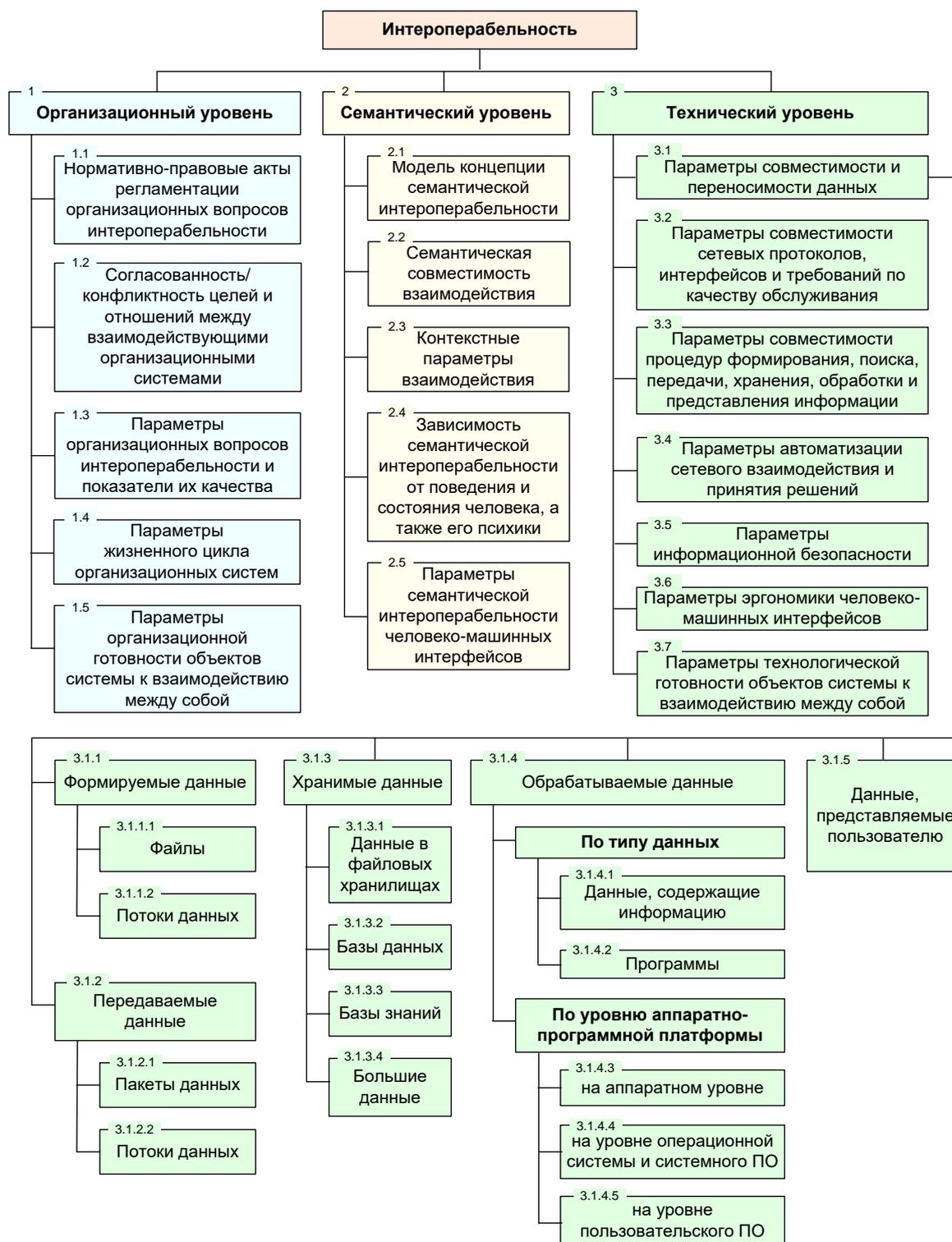


Рис. 1. Место и роль совместимости и переносимости данных при оценке технической интероперабельности СЦИУС

Ранее вопросы технической интероперабельности (блоки 3.1-3.7 на рис. 1) уже рассматривались в работах автора. В частности, параметры совместимости сетевых протоколов, интерфейсов и требований по качеству обслуживания (блок 3.2 на рис. 1) рассматривались в работе [15], параметры совместимости процедур

формирования, поиска, передачи, хранения, обработки и представления информации (блок 3.3 на рис. 1) – в работе [16], параметры автоматизации сетевого взаимодействия и принятия решений (блок 3.4 на рис. 1) – в работе [17], параметры информационной безопасности (блок 3.5 на рис. 1) – в работе [18], параметры эргономики человеко-машинных интерфейсов – в работах [19, 20]. Целью же данной работы является исследование вопросов совместимости и переносимости данных (блок 3.1 на рис. 1).

Исследование основано на известной модели «Systems, Capabilities, Operations, Programs, and Enterprises Model for Interoperability Assessment» (SCOPE-модели) [21], с учетом ее адаптации к отечественному подходу оценки интероперабельности, представленному в ГОСТ Р 55062-2012 [2]. Данное исследование продолжает и развивает ранее опубликованные работы автора [3-5, 15-20, 22-25], посвященные развитию отечественной теории интероперабельности.

1. Общие сведения о совместимости и переносимости данных

В общем случае под совместимостью и переносимостью в вычислительной технике понимается следующее.

Совместимость – способность объекта взаимодействовать и функционировать с другими объектами без каких-либо ограничений [26].

Переносимость – способность объекта относительно легко быть перенесенным с аппаратно-программной платформы одного типа на платформу другого типа [27].

По отношению к данным, можно дать следующее определение.

Совместимость данных – способность различных систем адекватно воспринимать и единообразно интерпретировать одинаково представленные данные.

Переносимость данных – способность данных относительно легко быть перенесенным с аппаратно-программной платформы одного типа на платформу

другого типа с сохранением их смысла и содержания, без необходимости их повторного ввода.

Данные – поддающееся многократной интерпретации представление информации в формализованном знаково-символьном виде, пригодном для сбора, хранения, передачи, обработки или представления в информационных системах [26].

Отметим, что понятия совместимости и переносимости данных можно рассматривать применительно к двум основным типам данных [28]:

1) данные, содержащие пользовательскую или полезную информацию в знаково-символьной форме;

2) данные, предназначенные для обработки данных первого типа – исходные тексты, исполнительные коды, сценарии или инструкции программ.

В работе [29] показано, что совместимость и переносимость данных, содержащих информацию, может быть обеспечена на основе трех основных подходов:

1) консолидация данных по принципу «выгрузка – преобразование – загрузка» – ETL (ExtractTransform-Load). При использовании подхода ETL исходные данные собираются из различных источников, полученные данные преобразовываются в соответствие с целевым форматом, переработанные данные загружаются в систему хранения данных;

2) интеграция данных в реальном времени с использованием федеративных запросов – ЕИ (Enterprise Information Integration). Подход ЕИ позволяет представлять множество источников и хранилищ данных в качестве единого виртуального пространства. При поступлении запроса пользователя, с помощью инструментов федерализации, осуществляется извлечение требуемых данных из источников и хранилищ данных, их обработка и интеграция в соответствие с правилами, определенными заранее, и представление результата пользователю;

3) синхронизация разнородных данных с помощью промежуточного программного обеспечения (ПО) – МОМ (Message-Oriented Middleware). Подход

МОМ предоставляет возможности синхронной или асинхронной передачи данных между системами управления базами данных (СУБД) и приложениями СЦИУС, выполняющими обработку данных.

Основными способами, с использованием которых может быть обеспечена переносимость и совместимость данных, являются [29, 30]:

1) обмен на основе файлов – наиболее распространенный способ обмена данными и взаимодействия приложений;

2) обмен на основе шлюзов данных – используется для доступа или обмена данными между различными базами данных (БД), при этом шлюз данных выполняет функции преобразования формата данных и запросов к ним, а также распределение данных между приложениями;

3) репликация данных – процесс обмена данными с целью приведения двух или более БД в идентичное состояние;

4) использование технологии web-сервисов, которые представляют собой удобное средство интеграции приложений, работающих с данными, и позволяют легко реализовать межплатформенное взаимодействие. Web-сервисы представляют собой объекты, находящиеся в определенных отношениях между собой, имеющие входные и выходные порты для входных и выходных данных и реализующие определенные функции обработки данных;

5) использование сервис-ориентированной архитектуры SOA (Service-Oriented Architecture). Система, реализованная по принципам SOA, представляет собой совокупность программных компонентов, а именно сервисов, имеющих стандартные интерфейсы для доступа к ним посредством сети и использования этих компонентов, при этом интерфейсы в SOA независимы от платформ развертывания сервисов и технологий их реализации;

6) использование архитектуры с управляемыми событиями EDA (Event Driven Architecture), которая является развитием архитектуры SOA и способна адаптироваться к потокам данных в бизнес-процессах. В EDA процессы могут выполняться как синхронно, так и асинхронно, при этом архитектура EDA может быть перепрограммируема при возникновении изменений в бизнес-процессах.

Кроме того, система, реализованная на основе EDA, способна воспринимать события из внешней среды;

7) использование технологии интеграционных серверов/брокеров, которые представляют собой промежуточные серверы в виртуальной среде данных, шлюзы, осуществляющие обработку потоков данных и запросов, а также распределение данных между приложениями, имеющими различные интерфейсы. В ядре такого сервера/брокера хранятся правила, на основе которых, а также полученных данных, выполняются вычислительные операции, анализ и принятие решений.

Следует иметь в виду, что переносимость и совместимость данных в реальных системах редко соответствует категориям «совместимость есть» / «совместимости нет» и «переносимость есть» / «переносимости нет». Чаще всего следует рассматривать категорию «обеспечение переносимости и совместимости с учетом затрат на реализацию». Для определения целесообразности реализации свойств переносимости и совместимости данных требуется анализ преимуществ и затрат.

Совместимость и переносимость данных в СЦИУС, прежде всего, обеспечиваются единообразием и универсальностью используемых форматов данных, как внутри системы, так и для взаимодействия с внешними системами. Для обеспечения этих свойств данных все компоненты СЦИУС должны обладать возможностью работать с теми форматами данных, которые используются в системе.

Формат данных – порядок расположения и форма представления данных в виде отдельных блоков, содержащих помимо непосредственно самих данных, еще и служебную информацию, облегчающую и упорядочивающую формирование, сбор, хранение, передачу, обработку или визуализацию данных [25].

Совместимость и переносимость данных – это важные свойства данных. Без обеспечения этих свойств, с одной стороны, не будет достигнута семантическая интероперабельность, т. к. сама возможность обмениваться

данными в некотором универсальном знаково-символьной формате подразумевает обмен информацией, с последующей однозначной интерпретации ее смысла (семантики), с другой стороны, эти свойства являются фундаментальными для обеспечения остальных аспектов технической интероперабельности, которые будут просто невозможными если не будет обеспечена совместимость и переносимость данных. Вследствие этого свойство обеспечения универсальности формата данных, в некоторых работах (например в [31, 32]) выделяют в отдельный вид интероперабельности – синтаксическую интероперабельность.

Синтаксическая интероперабельность – вид интероперабельности, позволяющей участвующим системам единообразно воспринимать и интерпретировать форматы информации/данных, которыми они обмениваются.

Синтаксическая интероперабельность, обеспечивается, прежде всего, единообразием форматов данных, для хранения целевой информации и едиными правилами структурирования данных, чтобы из формата данных можно было понять способ их передачи, хранения и обработки. Правила, определяющие структурирование данных должны позволять обнаруживать и справлять синтаксические ошибки формата в искаженных или неполных данных. При обеспечении переносимости данных синтаксис одного формата может быть преобразован в синтаксис другого формата. Там, где возможно точное преобразование синтаксиса, системы могут взаимодействовать без потерь. Примерами синтаксисов форматов данных могут быть форматы: HTML, XML, JSON, и синтаксисы, описанные средствами ASN.1 [31]. Способы синтаксического представления данных и их форматы должны изменяться по мере того, как технологическое развитие привносят новую функциональность или улучшают производительность службы. Вполне вероятно, что используемые форматы данных будут резко меняться в течение жизненного цикла системы. В целом, невозможно «заморозить» внутренние форматы данных таким образом, чтобы не препятствовать будущим инновациям. Лучше уже на этапе проектирования системы предусмотреть возможность перехода на новые

форматы данных, в случае если это потребуется для повышения функциональности системы.

Как будет показано далее, обеспечение универсальности формата данных, это важное, не единственное условие обеспечения совместимости и переносимости данных. Для обеспечения этих свойств необходимо обеспечить совместимость процедур на всем жизненном цикле данных.

В соответствии со стандартом [33] жизненный цикл данных включает в себя следующие этапы (рисунок 2):

- формирование данных;
- передача и сбор данных;
- хранение данных, в том числе и архивация, в интересах долговременного хранения данных;
- обработка данных (очищение, компиляция, преобразование, слияние, агрегирование и др.);
- использование данных – применение данных для решения различных пользовательских задач, главным образом, представление данных пользователю в том или ином виде в интересах информирования о состоянии целевых процессов, либо принятия управленческих решений;
- уничтожение (удаление) данных.

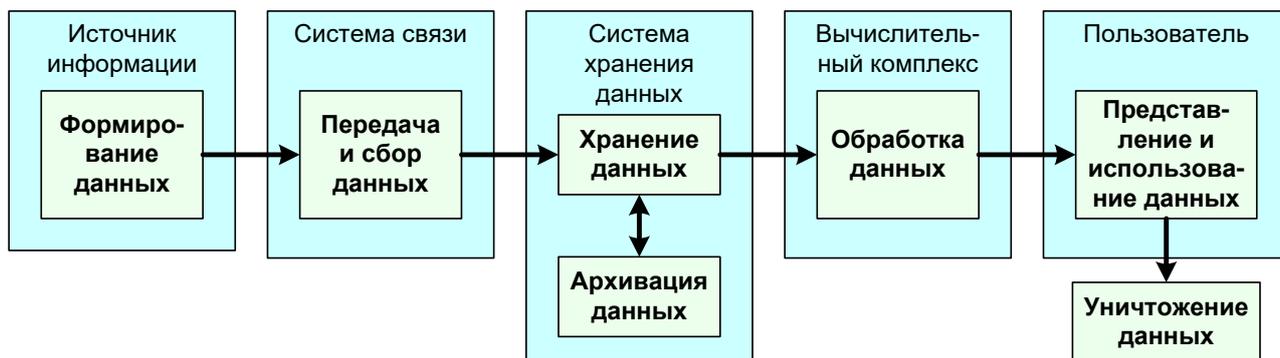


Рис. 2. Жизненный цикл данных

Процессы работы с данными на этапах жизненного цикла существенным образом зависят от особенностей реализации аппаратно-программных платформ, используемых в СЦИУС. В связи с этим требуется дополнительно

обеспечить совместимость и переносимость аппаратных и программных компонентов платформ, на которых реализуются этапы жизненного цикла данных.

Аппаратная совместимость – способность аппаратных и технических средств одной системы (платформы) функционировать и взаимодействовать с аналогичными средствами другой системы (платформы) [34].

Программная совместимость – способность программных средств одной системы (платформы) взаимодействовать и обмениваться данными с программными средствами другой системы (платформы) [34].

Аппаратная переносимость – возможность аппаратных и технических средств быть включенными в состав различных систем (платформ) с сохранением их работоспособности и качества функционирования [35]. Как правило аппаратная переносимость обеспечивается использованием различных специализированных аппаратно-зависимых модулей (драйверов) в составе ОС и системного ПО, которые изолируют от остальных программ особенности реализации информационных процессов в конкретных аппаратных средствах предоставляя программам универсальный типовой интерфейс взаимодействия.

Программная переносимость – возможность переноса программы из исходной системы (платформы) в другую систему (платформу) с сохранением ее функциональности в новой системе (платформе) [31].

Далее подробно рассмотрим вопросы обеспечения совместимости и переносимости данных на основных этапах жизненного цикла данных – формирование, передача, хранение, обработка и представление данных.

2. Формируемые данные

Для эффективного функционирования сетевой среды СЦИУС необходимо обеспечить сбор и использование данных обо всех участниках и объектах целевых процессов, их состоянии и параметрах, данных о состоянии окружающей среды и самой системы. Эти данные формируются источниками данных, которые могут представлять собой как технические средства, различной

природы (датчики, сенсоры, SCADA (Supervisory Control And Data Acquisition) системы, средства наблюдения и т.д.), так и людей – пользователей системы.

В общем случае источники данных можно разделить на:

- внутренние источники данных – отдельные элементы, подсистемы и компоненты СЦИУС;
- внешние источники данных – другие системы, внешние или временно подключаемые компоненты.

Источники данных могут формировать данные, соответствующие одному или несколькими типам информации: числовые значения неких параметров, значения логических переменных (true/false), символьные данные (текст), изображения, аудиоданные, видео, а также другие типы.

Формируемые данные могут быть представлены в следующих форматах:

- а) в виде отдельных файлов – в этом случае формат файла соответствует типу данных, которые в нем содержатся (числовые значения, текст, изображения, аудиоданные, видео и т.д.);
- б) в виде непрерывного потока данных, при этом такие потоки по своему формату могут быть:
 - структурированные – поток данных, имеет формат отдельных блоков, кадров, пакетов, структура которых заранее определена стандартами или протоколом формирования данных, что позволяет относительно легко обработать такой поток на приёмной стороне;
 - неструктурированные – поток данных, не имеет какого-либо стандартного формата и, как правило, представляет собой непрерывную последовательность байт.

Файл – совокупность данных определённого размера, размещённая на внешних устройствах хранения, рассматриваемая в процессе обработки как единое целое.

Формат файла – стандартизованная структура и способ организации данных для хранения их в файле.

Поток данных – последовательность непрерывно генерируемых данных. В отличие от файла поток данных не имеет какого-либо определенного размера или завершенной конечной реализации.

Формат потока данных – стандартизованная структура и способ организации данных в непрерывном потоке.

В современных СЦИУС данные как правило формируются источниками либо в виде файлов определенного типа (если источник данных имеет соответствующий объем собственной памяти), доступ к которым осуществляется удаленно по сетевой инфраструктуре системы, либо в виде структурированного потока данных, который передается по сети в место хранения или обработки (если источник данных не имеет собственной памяти для хранения формируемых данных).

Примерами форматов файлов, которые в настоящее время получили наибольшее распространения, являются:

- форматы структурированных данных – CSV, XMI, BIN, DAT, RAW, EDI, DICOM, SDXF и др.;
- форматы файлов документов – DOC, TXT, RTF, PDF, PS, ODT и др.;
- форматы файлов изображений без сжатия – PNG, TIFF, RAW и др.;
- форматы файлов изображений со сжатием – JPEG, BMP и др.;
- форматы файлов с аудиоданными – MP3, WMA, OGG, AAC, AC3, WAV, DTS, FLAC и др.;
- форматы файлов с видеоданными – AVI, MPEG, WMV, MKV и др.;
- форматы файлов с архивными сжатыми данными – ZIP, GZIP, LZX, RAR, 7Z и др.

Примерами форматов структурированных потоков данных, которые в настоящее время получили наибольшее распространения, являются – CSV, BIN, DAT, RAW и др.

Для корректного функционирования СЦИУС в ее составе должны использоваться только верифицированные источники данных, гарантирующие формирование данных, с требуемым качеством.

Качество данных – интегральный показатель, отражающий степень пригодности данных к решению задач пользователя.

В соответствии со стандартами [33, 36] качество данных оценивают по следующим частным показателям:

- *полнота данных* – данные отражают все ожидаемые характеристики исследуемых объектов в ожидаемом объеме; состав и объем данных достаточен для правильного понимания какого-либо явления или принятия решения;
- *достоверность данных* – соответствие данных реальному состоянию исследуемых или наблюдаемых объектов, фактов, событий или явлений;
- *точность данных* – степень искажения данных находится в допустимых пределах, а значения числовых параметров в пределах допустимой погрешности;
- *согласованность данных* – в данных отсутствуют внутренние противоречия, идентичные данные из различных источников совпадают;
- *целостность данных* – данные не были изменены при выполнении какой-либо операции (изменение, сбор, преобразование и т.д.);
- *обоснованность данных* – собранные данные отвечают поставленным целям и задачам;
- *расхождение данных во времени* – соответствие собираемых данных времени их возникновения;
- *уникальность данных* – в данных отсутствуют дубликаты;
- *валидность данных* – данные соответствуют ожидаемому формату, значения находятся в ожидаемых диапазонах и имеют ожидаемую точность.
- *доступность данных* – состояние данных, при котором субъекты, имеющие права доступа к данным, могут беспрепятственно получить доступ к данным;

- *своевременность данных* – предоставление доступа к данным в заданные сроки, когда данные сохраняют свою точность и могут быть использованы для принятия решений пользователями.

В процессе формирования данных необходимо избегать факторов, снижающих качество данных. К основным факторам, вызывающим снижение качества данных, относятся следующие: пропуск значений; наличие дублирования данных; противоречия в данных; аномальные значения и выбросы; шум; отсутствие полноты данных; нарушения целостности данных; некорректные форматы данных; фиктивные значения; ошибки ввода данных; нарушения структуры данных.

Для рациональной организации процедур формирования данных, в отношении их источников необходимо определить:

- цели и задачи, являющиеся основанием для формирования данных;
- требования к качеству формируемых данных;
- перечень, тип, формат и объем формируемых данных;
- способы и средства формирования данных;
- дальнейшие действия (передача, хранение, обработка и т.д.), которые предполагается выполнять с формируемыми данными.

Таким образом, для обеспечения совместимости и переносимости данных, формируемых различными источниками данных в СЦИУС, в рамках решения задачи обеспечения технической интероперабельности, необходимо:

- обеспечить совместимость типов передаваемой информации (цифровая, символьная, текст, изображение, аудио, видео);
- обеспечить совместимость и переносимость форматов файлов, в которые упаковывается формируемая информация;
- обеспечить совместимость и переносимость форматов потоков данных, в случае если формируемая информация немедленно передается без ее промежуточного хранения источником данных;

- обеспечить верификацию источников данных в системе, а также соответствие формируемых данных требованиям по качеству: по точности, по полноте, по согласованности и т.д.

3. Хранимые данные

Данные, формируемые источниками, передаются по сетевой инфраструктуре СЦИУС и по получении записываются в системы хранения данных, в интересах их долговременного хранения, либо для последующей обработки. Системы хранения данных могут представлять собой (рисунок 3): файловые хранилища, базы данных (БД), базы знаний (БЗ), хранилища больших данных. Структура организации и формат хранимых данных должен обеспечивать, во-первых, их синтаксическую совместимость со средствами обработки, с ПО доступа к данным со стороны пользователей и технических средств СЦИУС, во-вторых, переносимость данных между различными БД и БЗ.



Рис. 3. Классификация систем хранения данных

К организации хранения данных предъявляются следующие требования [33].

1) Основные данные должны иметь определенный набор *метаданных* – служебных данных для описания характеристик основных данных в БД, которые используются при организации процедур формирования, поиска, передачи, хранения, обработки и представления данных. Метаданные подразделяются на три основных категории:

- описательные метаданные – описывают содержание и состояние основных данных;

- технические метаданные – описывают технические особенности хранения основных данных;
- операционные метаданные – описывают особенности процессов обработки основных данных и доступ к ним.

2) Для основных данных и метаданных необходимо установить требования:

- по частоте обновления;
- по необходимости хранения архивных, более ранних версий и устаревших данных;
- по правам доступа к данным;
- по степени интеграции данных из различных источников;
- по процессам и правилам обновления данных;
- по ролям и обязанностям субъектов управления данными;
- по качеству данных.

3) Должно быть реализовано управление основными данными, включающее следующие мероприятия:

- установление сущностей и атрибутов основных данных;
- создание идентификаторов и перекрестных ссылок для интеграции данных из разных источников;
- объединение данных из различных источников для устранения несоответствий;
- дополнение и обновление основных данных.

4) Для обеспечения возможности замены и обновления системы хранения данных рекомендуется использовать открытые форматы хранения данных.

По отношению отдельной системе хранения данных может быть использован эквивалентный термин *информационный ресурс* – отдельный массив информации, который представлен в форме документов, массивов сведений, баз данных, баз знаний или других форм организованного представления информации [26]. Информационные ресурс может хранить структурированные или неструктурированные данные.

4.1. Файловые хранилища

Файловые хранилища могут быть организованы в виде выделенных файловых серверов или облачных хранилищ файлов. При этом в них могут храниться файлы произвольного формата.

Файловый сервер – это выделенный сервер в компьютерной сети, предназначенный для выполнения файловых операций ввода-вывода и хранения файлов любого типа.

Облачное хранилище файлов – децентрализованная распределённая совокупность файловых серверов, функционирующая под единым управлением (облако), позволяющее пользователям и приложениям получать доступ к файлам через совместно используемые файловые системы.

Таким образом, для обеспечения совместимости и переносимости данных в виде файлов необходимо обеспечить единообразие используемых форматов файлов, совместимость файловых систем файловых серверов или облачных хранилищ.

4.2. Базы данных

База данных представляет собой совокупность данных, характеризующих актуальное состояние некоторой предметной области, хранимых в компьютерных системах, при этом данные систематизированы и организованы в виде определенной логической структуры, снабжены дополнительными сведениями (атрибутами, указателями, ключами и т.д.) таким образом, чтобы данные могли быть найдены и обработаны с помощью средств вычислительной техники. Управление данными в БД (добавление, обновление, поиск, выбор, удаление данных), а также безопасность, журнализацию, надёжность хранения и целостность данных обеспечивает система управления базой данных (СУБД). Она же предоставляет средства для администрирования БД со стороны пользователя.

БД могут быть: реляционными или нереляционными; иерархическими и сетевыми; централизованными и распределёнными. При этом именно

декомпозиция реляционные и нереляционные БД является наиболее концептуальной.

Реляционные БД (от англ. relation – отношение, зависимость, связь) это «классические» БД, в которых хранятся структурированные данные, представленные в виде совокупности взаимосвязанных таблиц: строк – кортежей и столбцов – атрибутов. Для обслуживания реляционной БД и создания запросов к ней СУБД использует язык структурированных запросов SQL (Structured Query Language). Реляционные БД составляют порядка 90% всех используемых БД и идеально подходят для хранения именно структурированных данных, однако они не обладают должной степенью гибкости при хранении и работе с полуструктурированными или неструктурированными данными. Кроме того, по мере усложнения структуры организации данных становится все труднее обеспечить совместимость различных БД и переносимость данных между ними. Эти недостатки привели к развитию нереляционных БД, или, как их часто обозначают NoSQL.

Нереляционные БД получили свое развитие в 2010-х гг. при решении задач обработки полуструктурированных, неструктурированных и больших данных. В таких БД снижены требования к необходимости логической организации хранимых данных и встраивания их в какую-либо заранее заданную структуру, за счет чего эти БД могут хранить данные различного типа (текстовые данные, фото, видео, изображения и т.д.). Хранимые данные, которые снабжаются ключами, которые позволяют, задавая значения ключей, осуществлять поиск данных и формирование коллекций данных. В зависимости от модели данных, подходов к распределенности и репликации нереляционные БД можно декомпозировать на БД следующих основных типов [38]:

- БД типа «ключ – значение»;
- документно-ориентированные БД;
- столбцово-ориентированные БД;
- графовые БД.

С 2011 г. возник и постепенно получает распространение новый класс реляционных БД – NewSQL. БД NewSQL заимствуют от нереляционных БД оптимизацию обработки транзакций, масштабируемость, гибкость и бессерверную архитектуру, при этом сама структура организации данных в NewSQL соответствует реляционным БД. БД NewSQL могут масштабироваться, зачастую по запросу, не влияя на логику организации данных и не нарушая модель транзакций [38].

При обеспечении совместимости данных между различными БД и СУБД требуется не просто обеспечить совместимость форматов и типов данных, необходимо обеспечить совместимость моделей данных.

Модель данных – это описание представления БД и процессов обработки данных в СУБД, формализующее три основных аспекта:

- типы данных и структуру их организации;
- способы манипуляции с данными и выполнения операций над ними;
- способы описания и поддержки целостности БД.

Таким образом для обеспечения совместимости и переносимости данных на уровне БД и СУБД надо обеспечить:

- совместимость и переносимость типов и форматов данных;
- совместимость логических структур организации данных в различных БД;
- совместимость методов манипуляции с данными и выполнения операций над ними в различных СУБД;
- единство методов поддержки целостности БД со стороны СУБД.

4.3. Базы знаний

Данные, будучи обработаны интеллектуальными агентами в составе СЦИУС (более подробную информацию об интеллектуальных агентах см. в предыдущих работах автора [22, 23]), могут быть преобразованы в знания которые хранятся в *базах знаний* (БЗ) – базах, содержащих информацию о человеческом опыте, знаниях в некоторой предметной области, а также правила вывода из них новых знаний и решения задач [1].

Знания – совокупность информации о некоторой предметной области, хранящихся в формально-упорядоченном виде и пригодной для решения какой-либо задачи или достижения определенной цели; проверенный практикой и удостоверенный логикой результат познания действительности, отраженный в виде представлений, понятий, суждений и теорий [26]. В обобщенном виде можно записать: «знания» = «информация» + «цель» [40].

Предметные области в БЗ характеризуется индивидуальным набором понятий, законов, фактов, объектов и связей между ними. Способ структурного представления в БЗ набора понятий, законов, фактов, объектов и связей между ними называется онтологией.

Современные системы управления БЗ включают в себя подсистемы: поиска и извлечения информации; формирования новой информации и новых знаний; формирования новых решений на основе имеющейся информации.

В общем случае знания, содержащейся в БЗ, могут быть двух видов [40]:

- *процедурные знания* – последовательность действий, которое может использоваться при решении задач, достижения определенной цели и получении новых знаний (процедурные знания могут быть реализованы, например, в виде совокупности программ);
- *декларативные знания* – знания в форме описания фактов, событий или явлений, к которым носитель знания имеет осознанный доступ и которое он может декларировать – т.е. изложить в вербальной (языковой) или знаково-символьной (письменной) форме.

Знания в БЗ могут быть логически организованны в соответствии с одной из моделей знаний (рис. 4) [40]:

- производственная модель знаний;
- модель знаний типа «семантическая сеть»;
- фреймовая модель знаний;
- логическая модель знаний;
- псевдофизическая модель знаний.

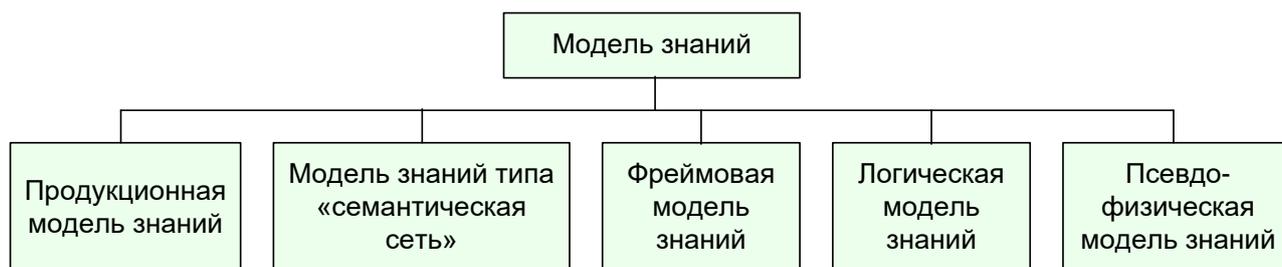


Рис. 4. Типы модели знаний

Более подробные сведения о вышеуказанных моделях знаний представлены в работе [40].

При обеспечении совместимости и переносимости знаний между различными БЗ и системами управления БЗ требуется обеспечить:

- совместимость и переносимость типов знаний (процедурные и декларативные);
- использование БЗ модели знаний одного и того же типа, а также совместимость структур организации информации в них;
- совместимость методов обработки информации, процедур формирования новой информации и новых знаний, а также новых решений на основе имеющейся информации;
- совместимость и переносимость предметных областей, к которым относятся знания и информация;
- совместимость вида представления и формата информации, хранимой в БЗ.

4.4. Большие данные

Большие данные – совокупность данных обладающих следующими ключевыми характеристиками (так называемые «три V»): volume – чрезвычайно большим объёмом; velocity – как высокой скоростью прироста, так и необходимостью высокоскоростной их обработки для получения результатов; variety – значительного многообразия данным, сочетания структурированных, полуструктурированных и неструктурированных данных, которые должны обрабатываться совместно [41].

В качестве примеров источников больших данных в СЦИУС можно привести: измерительные устройства и средства наблюдения, непрерывно генерирующие данные; подсистемы обмена сообщениями между пользователями в сети; источники метеорологических данных и данных дистанционного зондирования Земли; навигационные устройства, генерирующие данные о местонахождении объектов и пользователей системы и т.д.

Современные возможности технических средств обработки позволяют провести анализ вышеуказанных больших данных в интересах выявления закономерностей развития процессов, типовых паттернов и состояний элементов системы, взаимовлияния компонентов системы друг на друга, прогнозирования поведения системы при изменении условий или внешней среды. Фактически, технология хранения и обработки больших данных находится на стыке использования нереляционных БД типа NoSQL для хранения данных и методов обработки и извлечения новой информации, заимствованных из систем управления БЗ. Для практической реализации технологии больших данных используются нереляционные БД, языки программирования для статистической обработки и анализа данных (сейчас широко используется язык R), а также система распределенных параллельных вычислений на кластерной инфраструктуре (сейчас широко используется фреймворки Hadoopи MapReduce).

При обеспечении совместимости и переносимости больших данных требуется обеспечить:

- совместимость и переносимость форматов структурированных, полуструктурированных и неструктурированных данных;
- совместимость моделей данных нереляционных БД в которых хранятся большие данные;
- совместимость методов статистической обработки и анализа данных, методов формирования и извлечения новой информации;
- совместимость аппаратных и программных компонентов платформы распределенных параллельных вычислений, используемых для обработки больших данных.

5. Обработываемые данные

Данные, поступающие от источников данных или хранимые в системах хранения данных, передаются на обработку. Обработка данных представляет собой совокупность операций над данными, совершаемых с использованием средств автоматизации или вычислительной техники, направленных на повышение качества данных. К таким операциям относятся: очистка, агрегирование, систематизация, инвентаризация, маркировка, сортировка, верификация, уточнение, обогащение, обновление, поиск, изменение, извлечение, использование, обезличивание и т.д.

К организации обработки данных предъявляются следующие требования [33].

1) Если сформированные или собранные данные являются первичными, то перед их обработкой необходимо провести их очистку и верификацию.

2) Данные после очистки и верификации для достижения установленного уровня качества объединяют в наборы данных для дальнейшего использования посредством операций агрегирования, систематизации, маркировки, сортировки и уточнения. На этом этапе возможно формирование новых данных, являющихся выводами по результатам обработки первичных данных.

3) Для каждой отдельной разновидности организации данных должен быть определен формат набора данных.

4) Поскольку качество данных может снизиться на любом этапе жизненного цикла данных, необходимо планировать меры по обеспечению качества данных в расчете на весь их жизненный цикл.

5) После обновления данных целесообразно повторно провести мероприятия обработки, направленные на повышение качества данных и корректного функционирования системы обработки данных СЦИУС.

Определение качества данных и основные его частные показатели представлены в подразделе 2 настоящей статьи.

Как правило, данные обрабатываются в вычислительных комплексах, представляющих собой трехуровневую аппаратно-программную платформу (рисунок 5).

Аппаратно-программная платформа – совокупность вычислительных аппаратных средств и, функционирующих поверх них, ОС и системного ПО, работающих совместно для выполнения пользовательских задач [43].



Рис. 5. Структура вычислительного комплекса обработки данных [27, 35]

С учетом вышеуказанной трехуровневой компоновки вычислительного комплекса для обеспечения полной совместимости и переносимости данных эти свойства требуется обеспечить на всех его уровнях.

5.1. Уровень аппаратных средств

Особенностью уровня аппаратных средств является то, что на нем рассматриваются два основных типа данных:

- непосредственно сами данные, требующие обработки;
- команды и инструкции, кодирующие операции, выполняемые над данными.

В связи с этим на уровне аппаратных средств требуется обеспечить:

- аппаратную переносимость вычислительных средств/модулей между различными вычислительными комплексами/платформами в составе СЦИУС;
- аппаратную совместимость используемых процессоров, объединенных в рамках одной платформы;
- совместимость разрядности и формата временного хранения данных в оперативной памяти;
- совместимость разрядности, форматов команд и инструкций вычислительных средств/модулей (процессоров), используемых в рамках одной платформы;
- совместимость всех внешних устройств по возможностям их совместной и одновременной работы в рамках одной платформы.

5.2. Уровень операционной системы и системного программного обеспечения

Особенностью уровня ОС и системного ПО является то, что ОС может путем реализации нескольких аппаратно-зависимых модулей ОС и технологии множественных прикладных сред обеспечить обработку данных различными программами на базе различных аппаратных средств (рисунок 6). В этом случае

ОС изолирует особенности выполнения отдельных программ и функционирования аппаратных средств друг от друга беря на себя функцию транслятора системных вызовов и данных между всеми аппаратными и программными компонентами, взаимодействующими между собой в составе вычислительного комплекса [27, 35].

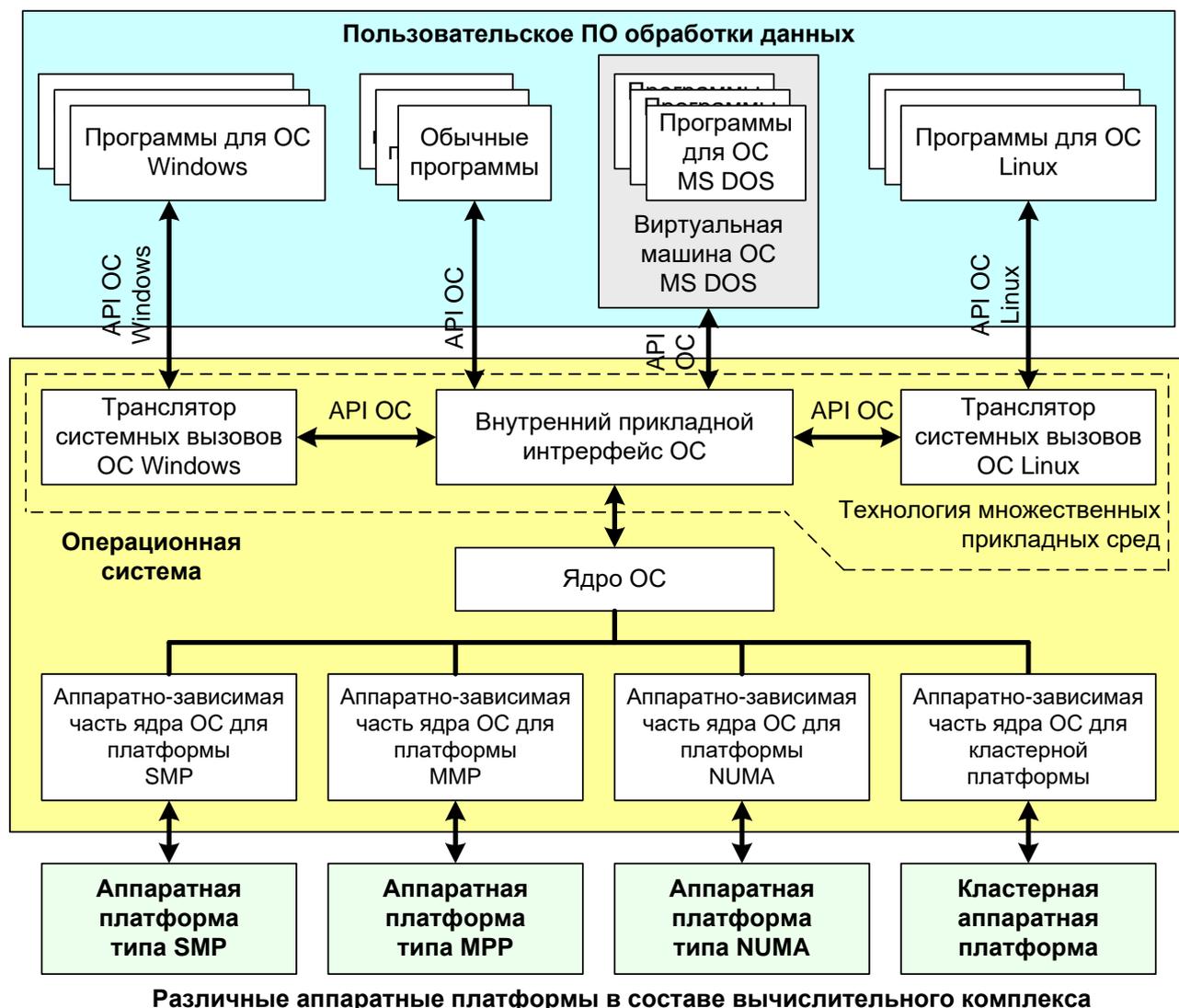


Рис. 6. Вариант реализации нескольких аппаратно-зависимых модулей ОС и технологии множественных прикладных сред в интересах обработки данных различными программами на базе различных аппаратных средств [27, 35]

Для совместимости и переносимости данных на уровне ОС и системного ПО требуется обеспечить:

- реализацию аппаратно-зависимых частей ядра ОС для всех аппаратных платформ, используемых в составе вычислительного комплекса;

- реализацию аппаратно-зависимых модулей ОС – драйверов, правил ввода/вывода, сетевых протоколов и т.д., для всех аппаратных средств, используемых для обмена данными;
- реализацию в составе ОС технологии множественных прикладных сред, в которой требуется обеспечить поддержку тех интерфейсов прикладного программирования API (Application programming interface), которые используются пользовательским ПО обработки данных; аппаратно-зависимых частей ядра ОС для всех аппаратных платформ, используемых в составе вычислительного комплекса;
- для расширения совместимости ПО, запуска унаследованного и устаревшего ПО обработки данных целесообразным является внедрение в состав системного ПО вычислительного комплекса виртуальных машин, в которых установлены соответствующие ОС.

5.3. Уровень пользовательского программного обеспечения

Для совместимости и переносимости данных на уровне пользовательского ПО требуется обеспечить:

- совместимость пользовательского ПО с максимально широким спектром форматов файлов, БД и БЗ, используемых в СЦУИС;
- программную переносимость пользовательского ПО во все основные ОС, используемые в аппаратно-программных комплексах СЦУИС;
- совместимость инструкций и процедур обработки данных в различных программах пользовательского ПО;
- универсальность системы команд в интересах формирования сложных сценариев обработки данных, в которых задействованы различные программы пользовательского ПО;
- совместимость служб обмена данными (например, SOA или «клиент – сервер»), используемыми различными программами обработки данных;
- написание пользовательского ПО на тех языках программирования, которые позволяют компилировать конечные программы во все

основные ОС, используемые в аппаратно-программных комплексах СЦУИС;

- использование открытого исходного кода для создания пользовательского ПО.

Наиболее важным из вышеперечисленного является совместимость пользовательского ПО с максимальным числом форматов данных, программная совместимость и переносимость пользовательского ПО.

Относительно формата программ отметим следующее. Программы представляют собой последовательность инструкций и команд по обработке данных. Однако помимо команд, программы получают, хранят и обрабатывают как целевые данные, так и свои служебные данные, необходимые программам для их корректной работы. Структура и формат хранения этих данных определяется разработчиком программы. Даже две версии одной и той же программы могут разрабатываться по-разному, а также хранить данные совершенно по-разному. С учетом этого, программная переносимость пользовательского ПО между ОС с учетом особенностей формата программ может быть обеспечена двумя основными способами [31]:

- обеспечение синтаксической совместимости программ с ОС – представление программ в таком формате, который может быть непосредственно декодирован, запущен и выполнен в целевой ОС;
- обеспечение командной совместимости программ с ОС – команды программ могут выполняться посредством их трансляции в команды API целевой ОС посредством реализации технологии множественных программных сред или с использованием виртуальной машины.

Помимо реализации указанных способов важным для обеспечения переносимости пользовательского ПО является совместимость служебных данных программы, которая подразумевает что программа может получить доступ к служебным данным, необходимым ей для выполнения, и одинаковым образом декодировать и интерпретировать их при работе в различных ОС [31].

6. Данные, предоставляемые пользователю

При использовании данных, как правило, они предоставляются пользователю в графическом/визуальном виде, реже – в виде речевых сообщений или тактильно-воспринимаемых операций (например, посредством передачи усилия обратной связи на штурвал самолета). Данные представляются для того чтобы пользователь получил полную и достоверную информацию о целевых процессах, либо в интересах принятия пользователем решения.

Представление данных пользователю осуществляется посредством человеко-машинных интерфейсов (ЧМИ). Вопросы организации эргономических ЧМИ, их влияние на эффективность функционирования всей системы, а также на повышение уровня информационного взаимодействия пользователя с технической системой, подробно были рассмотрены в предыдущих работах автора [19, 20] и здесь не рассматриваются.

Заключение

На основании вышеизложенного можно сделать следующие выводы.

1) Одной из основных составляющих технической интероперабельности СЦИУС является совместимость и переносимость данных.

2) Понятия совместимости и переносимости данных нужно рассматривать применительно к данным двух типов:

- данные, содержащие пользовательскую или полезную информацию в знаково-символьной форме;
- данные, предназначенные для обработки данных первого типа – исходные тексты, исполнительные коды, сценарии или инструкции программ.

3) Для обеспечения полной совместимости и переносимости данных эти свойства должны реализовываться на всех этапах жизненного цикла данных, с учетом особенностей использования в СЦИУС различных аппаратно-программных платформ:

- на этапе формирования данных;

- на этапе передачи и сбора данных;
- на этапе хранения данных;
- на этапе обработки данных;
- на этапе представления и использования данных.

4) Рассмотренные в статье аспекты обеспечения совместимости и переносимости данных целесообразно в дальнейшем использовать при разработке проекта рекомендаций по обеспечению интероперабельности СЦИУС, дополняющих ГОСТ Р «Информационные технологии. Сетецентрические информационно-управляющие системы. Интероперабельность».

Финансирование. Результаты, представленные в данной работе, получены в рамках госбюджетной темы НИР FFZF-2022-0004.

Литература

1. ISO/IEC/IEEE 24765:2017. *Systems and software engineering. Vocabulary*. ISO. 2017. 522 p.
2. ГОСТ Р 55062-2012. *Информационные технологии (ИТ). Системы промышленной автоматизации и их интеграция. Интероперабельность. Основные положения*. Москва, Стандартинформ. 2014. 12 с.
3. Башлыкова А.А., Козлов С.В., Макаренко С.И., Олейников А.Я., Фомин И.А. Подход к обеспечению интероперабельности в сетецентрических системах управления. *Журнал радиоэлектроники* [электронный журнал]. 2020. №6. <https://doi.org/10.30898/1684-1719.2020.6.13>
4. Макаренко С.И., Олейников А.Я., Черницкая Т.Е. Модели интероперабельности информационных систем. *Системы управления, связи и безопасности*. 2019. №4. С.215-245. <https://doi.org/10.24411/2410-9916-2019-10408>

5. Козлов С.В., Макаренко С.И., Олейников А.Я., Растягаев Д.В., Черницкая Т.Е. Проблема интероперабельности в сетевых системах управления. *Журнал радиоэлектроники* [электронный журнал]. 2019. №12. <https://doi.org/10.30898/1684-1719.2019.12.4>
6. Маслобоев А.В. Средства поддержки интероперабельности сетевых систем управления региональной безопасностью. *Надежность и качество сложных систем*. 2020. №1 (29). С.91-105. <https://doi.org/10.21685/2307-4205-2020-1-11>
7. Маслобоев А.В. Проблемы и технологии обеспечения интероперабельности информационных систем региональных ситуационных центров. *Информационно-технологический вестник*. 2020. №2 (24). С.107-119.
8. Осипенков М.Н., Узякаев И.Н. Основные проблемы достижения интероперабельности информационных систем органов государственного и военного управления при решении задач обороны. *Военная мысль*. 2020. №5. С.143-149.
9. Куприянов А.А. Аспекты интероперабельности автоматизированных систем. *Автоматизация процессов управления*. 2009. №4. С.40-49.
10. Куприянов А.А. Сетевые военные действия и вопросы интероперабельности автоматизированных систем. *Автоматизация процессов управления*. 2011. №3. С.82-97.
11. Башлыкова А.А., Каменщиков А.А., Олейников А.Я. Обеспечение интероперабельности как средства бесшовной интеграции функциональных подсистем в составе перспективных автоматизированных систем военного назначения. *Журнал радиоэлектроники* [электронный журнал]. 2018. № 9. С.18. <https://doi.org/10.30898/1684-1719.2018.9.11>
12. Каменщиков А.А., Олейников А.Я., Чусов И.И., Широкова Т.Д. Проблема интероперабельности в информационных системах военного назначения. *Журнал радиоэлектроники* [электронный журнал]. 2016. №11. С.16. <http://jre.cplire.ru/jre/nov16/8/text.pdf>

13. Зацаринный А.А., Козлов С.В. Процессные аспекты нормативного регулирования работ по комплексному обеспечению информационной безопасности и интероперабельности интегрированных систем управления. *Информатика: проблемы, методы, технологии. Материалы XXI Международной научно-методической конференции*. Воронеж, Вэлборн. 2021. С.1167-1176.
14. Башлыкова А.А., Зацаринный А.А., Каменщиков А.А., Козлов С.В., Олейников А.Я., Чусов И.И. Интероперабельность как научно-методическая и нормативная основа бесшовной интеграции информационно-телекоммуникационных систем. *Системы и средства информатики*. 2018. Т.28. №4. С.61-72. <https://doi.org/10.14357/08696527180407>
15. Макаренко С.И., Черницкая Т.Е. Аспекты совместимости сетевых протоколов, интерфейсов и требований по качеству обслуживания в рамках оценки интероперабельности сетевых информационных управляющих систем. *Журнал радиоэлектроники* [электронный журнал]. 2020. №10. С.7. <https://doi.org/10.30898/1684-1719.2020.10.4>
16. Макаренко С.И. О некоторых параметрах поиска и обработки информации при обеспечении технической интероперабельности сетевых информационных систем. *Журнал радиоэлектроники* [электронный журнал]. 2021. №3. <https://doi.org/10.30898/1684-1719.2021.3.5>
17. Черницкая Т.Е., Макаренко С.И., Растягаев Д.В. Аспекты автоматизации функций управления, принятия решений и сетевого взаимодействия в рамках оценки интероперабельности сетевых информационных управляющих систем. *Вестник Российского нового университета. Серия: Сложные системы: модели, анализ и управление*. 2020. №3. С.138-145. <https://doi.org/10.25586/RNU.V9187.20.03.P.138>

18. Черницкая Т.Е., Макаренко С.И., Растягаев Д.В. Аспекты информационной безопасности в рамках оценки интероперабельности сетевых информационных-управляющих систем. *Вестник Российского нового университета. Серия: Сложные системы: модели, анализ и управление*. 2020. №4. С.113-121. <https://doi.org/10.25586/RNU.V9187.20.04.P.113>
19. Макаренко С.И. Семантическая интероперабельность человеко-машинных интерфейсов в сетевых системах. *Журнал радиоэлектроники [электронный журнал]*. 2022. №1. <https://doi.org/10.30898/1684-1719.2022.2.4>
20. Макаренко С.И. Техническая интероперабельность и эргономика человеко-машинных интерфейсов сетевых систем. *Журнал радиоэлектроники [электронный журнал]*. 2022. №3. <https://doi.org/10.30898/1684-1719.2022.3.4>
21. *Systems, Capabilities, Operations, Programs, and Enterprises (SCOPE) Model for Interoperability Assessment. Version 1.0*. NCOIC. 2008. 154 p.
22. Макаренко С.И., Соловьева О.С. Основные положения концепции семантической интероперабельности сетевых систем. *Журнал радиоэлектроники [электронный журнал]*. 2021. №4. <https://doi.org/10.30898/1684-1719.2021.4.10>
23. Макаренко С.И., Соловьева О.С. Семантическая интероперабельность взаимодействия элементов в сетевых системах. *Журнал радиоэлектроники [электронный журнал]*. 2021. №6. <https://doi.org/10.30898/1684-1719.2021.6.3>
24. Макаренко С.И. Семантическая совместимость человеческих агентов при обеспечении интероперабельности в сетевых системах. *Журнал радиоэлектроники [электронный журнал]*. 2022. №1. <https://doi.org/10.30898/1684-1719.2022.1.1>
25. Макаренко С.И., Карутин А.Н. Перспективы и проблемные вопросы обеспечения интероперабельности интегрированных космических систем. *Системы управления, связи и безопасности*. 2021. №4. С.228-247. <https://doi.org/10.24412/2410-9916-2021-4-228-247>

26. Макаренко С.И. *Справочник научных терминов и обозначений*. Санкт-Петербург, Научное издание. 2019. 254 с.
27. Макаренко С.И., Ковальский А.А., Краснов С.А. *Принципы построения и функционирования аппаратно-программных средств телекоммуникационных систем: учебное пособие. Часть 2: Сетевые операционные системы и принципы обеспечения информационной безопасности в сетях*. Санкт-Петербург, Научное издание. 2020. 357 с.
28. Табаков Ю. Переносимость данных и программ [WEB]. *C++ Builder. Интернет учебник для всех*. Дата обращения: 11.05.2022. URL: <https://cubook.pro/perenosimost-dannykh-i-programm>
29. Прокофьева И.В., Шибанов С.В., Шашков Б.Д. Анализ применения современных технологий интеграции данных в разнородных распределенных информационных системах. *Труды международного симпозиума "Надежность и качество"*. 2009. Т.1. С.243-249.
30. Шибанов С.В., Яровая М.В., Шашков Б.Д., Кочегаров И.И., Трусов В.А., Гришко А.К. Обзор современных методов интеграции данных в информационных системах. *Труды международного симпозиума "Надежность и качество"*. 2010. Т.1. С.292-295.
31. ГОСТ Р ИСО/МЭК 19941-2021. *Информационные технологии. Облачные вычисления. Интероперабельность и переносимость*. Москва, Стандартинформ. 2022. 74 с.
32. Батоврин В.К., Гуляев Ю.В., Олейников А.Я. Обеспечение интероперабельности - основная тенденция в развитии открытых систем. *Информационные технологии и вычислительные системы*. 2009. №5. С.7-15.
33. ГОСТ Р 59897-2021. *Данные для систем искусственного интеллекта в образовании. Требования к сбору, хранению, обработке, передаче и защите данных*. Москва, Стандартинформ. 2021. 7 с.
34. Воройский Ф.С. *Информатика. Новый систематизированный толковый словарь-справочник*. Москва, ФИЗМАТЛИТ. 2003. 760 с.

35. Макаренко С.И. *Вычислительные системы, сети и телекоммуникации*. Ставрополь, Ставропольский филиал МГГУ имени М.А. Шолохова. 2008. 352 с.
36. ISO 9000:2015. *Quality management systems - Fundamentals and vocabulary*. International Organization for Standardization, 2015. 51 p.
37. Макаренко С.И., Федосеев В.Е. *Системы многоканальной связи. вторичные сети и сети абонентского доступа*. Санкт-Петербург, Военно-космическая академия им. А.Ф. Можайского. 2014. 179 с.
38. Ключевые аспекты при выборе базы данных для вашего приложения [WEB]. *Habr.ru*. Дата обращения: 11.05.2022. URL: <https://habr.com/ru/company/otus/blog/562852/>
39. Harkushko L. Vital things to consider when choosing a database for your App [WEB]. *Yalantis*. Дата обращения: 11.05.2022. URL: <https://yalantis.com/blog/how-to-choose-a-database/>
40. Макаренко С.И. *Интеллектуальные информационные системы*. Ставрополь, Ставропольский филиал МГГУ им. М. А. Шолохова. 2009. 206 с.
41. Майер-Шенбергер В., Кукьер К. *Большие данные. Революция, которая изменит то, как мы живём, работаем и мыслим*. Москва, Манн, Иванов, Фербер. 2014. 240 с.
42. Manyika J. et al. *Big data: The next frontier for innovation, competition, and productivity*. McKinsey Global Institute. 2011. <https://www.mckinsey.com/mgi/our-research/all-research>
43. Белоножко П.П., Белоус В.В., Куцевич Н.А., Храмов Д.А. Свободныеоблачные аппаратно-программные платформы. Аналитический обзор. *Интернет-журнал «Науковедение»*. 2016. №6 (8). <https://doi.org/10.15862/61TVN616>

Для цитирования:

Макаренко С.И. Совместимость и переносимость данных при обеспечении технической интероперабельности сетевых систем. *Журнал радиоэлектроники* [электронный журнал]. 2022. №7. <https://doi.org/10.30898/1684-1719.2022.7.1>