

УДК 004.7

ПОДХОДЫ К РЕШЕНИЮ ПРОБЛЕМ ПРОЗРАЧНОСТИ РАСПРЕДЕЛЕННЫХ ИНФОРМАЦИОННЫХ СИСТЕМ

Лещев С.А., студент гр. ПИМ-191, 1 курс
Научный руководитель: Крюкова В.В., к.т.н., доцент
Кузбасский государственный технический университет
имени Т.Ф. Горбачева

В современном мире число устройств, обменивающихся друг с другом информацией, неуклонно растёт. Если в конце 90–х – начале 2000–х большинство персональных компьютеров даже не имело доступ в Интернет, то сейчас компьютер, не имеющий доступ к всемирной паутине, для многих пользователей не представляет никакой ценности. По мимо этого за последние десять лет прочно укоренилось такое явление как IoT (Internet of Things – Интернет вещей). По данным Strategy Analytics число устройств, подключённых к сети, к концу 2018 г. оценивалось в 22 млрд., при этом в прогнозе на 2025 г. эта цифра достигает 38 млрд. устройств [1]. Вместе с ростом количества устройств, подключённых к сети, растёт и роль распределённых информационных систем (РИС), в которых работают эти устройства.

Основной проблемой при переходе от работы отдельного устройства к работе этого устройства в распределённой системе, является создание иллюзии того, что устройство (или конечный пользователь) работает с монолитной системой. Свойство РИС создавать видимость работы как единой системы называется прозрачностью и является ключевым требованием при разработке распределённой системы.

Целью настоящей работы является анализ обеспечения требования прозрачности РИС на этапах создания и эксплуатации.

В статье рассматриваются:

- проблемы обеспечения прозрачности на этапах создания и эксплуатации;
- подходы к решению проблем прозрачности на этапах проектирования и эксплуатации.

Основные проблемы обеспечения прозрачности в РИС [2] можно свести к трём группам:

1. Проблема доступа к ресурсу – в распределённой системе необходимо обеспечивать постоянный доступ к ресурсам, размещённым на удалённых узлах. При этом возникают следующие трудности:

- ресурс может перемещаться между узлами, в том числе в момент непосредственного обращения к нему пользователя – прозрачность перемещения;

- сразу несколько пользователей могут обращаться к одному ресурсу или копиям одного ресурса – прозрачность параллельного доступа, прозрачность репликации.

2. Проблема открытости распределённой системы – поскольку РИС включает в себя множество узлов, это приводит к тому, что:

- узлы могут по-разному представлять одни и те же данные, что может искажать информацию при информационном обмене – прозрачность доступа;
- информация может храниться на разных устройствах и носителях, при этом пользователю необходимо обеспечить возможность не задумываться, с какого именно устройства он получает данные – прозрачность сохранности.

3. Проблема отказоустойчивости РИС – большое число элементов распределённой системы и их географическая удаленность увеличивают вероятность выхода из строя одного узла или части системы, поэтому в РИС возникают проблемы:

- обеспечения доступа к ресурсу, если узел, на котором ресурс размещён, «выпал» из распределённой системы – прозрачность отказов;
- синхронизации узла или части системы, после того как они вернулись в строй.

Одним из основных подходов при проектировании РИС является принцип декомпозиции [3], это позволяет выделить специфические функции системы в отдельные узлы, что, безусловно, улучшает масштабируемость системы, однако усложняет связь её компонентов между собой. Для инкапсуляции внутренних различий узлов необходимо проектировать application programming interface (API-интерфейсы), что позволит обращаться к ресурсу используя лишь открытые методы и свойства и не даст возможности использовать различающиеся свойства отдельных узлов.

Разделение системы на отдельные компоненты даёт возможность для резервирования каждого компонента в отдельности. Это позволяет уделить больше внимание на резервировании критически важных компонентов РИС, но при этом сэкономить на резервировании менее важных. Однако при создании реплик ресурсов появляется необходимость обеспечивать прозрачность репликации.

Показателями качества репликации являются:

- RPO (recovery point objective) – допустимая потеря данных (минут);
- RTO (recovery time objective) – допустимое время восстановления данных (минут).

Суть проблемы прозрачности репликации в сложности обеспечить RPO = 0 и RTO = 0. Компании предлагают различные варианты решения данной проблемы, например, синхронная репликация хранилищ данных от HP или «горячее резервирование» СУБД Oracle. Однако, данные методы ограничены максимальным расстоянием, которое определяется максимально допустимым для синхронной репликации временем задержки между двумя узлами (приблизительно 10 мс.). Данное ограничение не позволяет разносить реплицируемые узлы более чем на 100 км.

Одним из вариантов решения данной проблемы является использование синхронной репликации между двумя узлами А и В совместно с асинхронной репликацией между узлами А, В и узлом С [4].

Помимо резервирования при сбоях, репликация ресурса позволяет сделать его более доступным и повысить скорость обращения к нему. Такой эффект достигается за счет более близкого расположения ресурса к пользователю. Однако, в этом случае возникает проблема параллельного доступа к репликам ресурса нескольких пользователей суть которой состоит в том, что изменения, произошедшие раньше по времени должны быть внесены в систему и отобразиться во всех репликах ресурса для всех пользователей до того, как будут внесены изменения, произошедшие позже по времени. То есть, в распределённой системе не должен нарушаться принцип причинности.

На этапе проектирования эта проблема решается выбором оптимальной топологии сети передачи данных, выбором аппаратных средств, способных обеспечивать высокую скорость обработки данных, применением скоростных протоколов и каналов передачи данных (например, использование оптоволокон), синхронизация времени всех узлов РИС по GPS.

Подходы к обеспечению прозрачности, заложенные на этапе проектирования РИС, должны так же подкрепляться на этапе её эксплуатации, максимально используя возможности программно-аппаратных средств и топологию системы.

В процессе эксплуатации РИС необходимо обеспечивать доступ к ресурсу без указания его физического местоположения, поскольку он может перемещаться от узла к узлу. Несмотря на то, что URL является логическим именем, он указывает на место расположения ресурса, и, в случае его переименования, удаления, перемещении в другой домен DNS, URL становится не действительным. Для устранения недостатков, присущих URL, были предложены постоянные указатели PURL (Persistent Uniform Resource Locator). В сущности, это те же URL, но они указывают не на конкретное место расположения ресурса, а на запись в базе данных PURL, где, в свою очередь, записан уже конкретный URL -адрес ресурса. Однако, использование PURL лишь частичное решения проблемы. Принципиальным же решением должен стать стандарт единообразного именования ресурсов URN (Uniform Resource Name – единообразное имя ресурса) [5], указывающий неизменное имя ресурса без указания его местонахождения и способа обращения. В результате URN–имена постоянны, они не зависят от конкретных серверов и протоколов.

Поскольку пользователи могут обращаться к ресурсу при его перемещении, то при изменении местоположения ресурса важно обеспечивать также прозрачность перемещения. Реализация прозрачности перемещения в распределённой системе может оказаться довольно сложной процедурой. Иногда ею целесообразно пренебречь, если перемещение ресурсов происходит редко (например, можно отключить ресурс от сети и выполнить перемещение в офлайн) [6].

Если же к ресурсу обращается много пользователей, или перемещение ресурсов происходит часто, то пренебрежение данной формой прозрачности приведет к тому, что распределенная система перестанет казаться единым целым. Для конечного пользователя она будет не совсем прозрачной, что может сказаться на самоцели создания распределенной системы.

Одним из способов обеспечения прозрачности перемещения является следующая последовательность действий:

- 1) создание копии ресурса;
- 2) дублирование канала связи от пользователя до копии ресурса
- 3) обеспечение синхронной репликации;
- 4) отключение первичного канала связи до первоначального ресурса и переключение пользователя на работу с копией ресурса.

Таким образом, частые перемещения ресурсов между узлами РИС могут существенно замедлить скорость её работы, поэтому необходимо заранее планировать размещение реплик ресурса на узлах, наиболее приближенных к конечному пользователю.

Прозрачность отказов на этапе эксплуатации РИС обеспечивается применением алгоритмов обработки исключительных ситуаций и синхронизации данных [7]:

- отказоустойчивые алгоритмы - гарантируют, что, несмотря на сбои в некоторых процессах (узлах), неповреждённые процессы (узлы) корректно выполняют свои функции. Устойчивые алгоритмы защищают систему против отказов ограниченного числа узлов. Остающиеся работоспособными узлы поддерживают правильное (хотя менее эффективное) поведение во время восстановления и реконфигурации системы. Такие алгоритмы должны использоваться, когда невозможно временное прерывание работы.

- стабилизирующие алгоритмы - гарантирует исправление ошибок в случае сбоев по причине временного достижения критических значений, что приводит к ошибочному поведению узлов. Свойство стабилизации гарантирует сходимость к требуемому поведению системы после окончания действия на систему негативных факторов. Примером стабилизирующих алгоритмов являются: алгоритмы самовосстановления, отказоустойчивое кэширование, логика повторения.

Таким образом, обеспечение прозрачности распределения является комплексной задачей, которая должна решаться на всех этапах проектирования и эксплуатации распределённых систем. На наш взгляд, наибольшее внимание следует уделить проблемам прозрачности репликации и параллельного доступа. Поскольку данные аспекты прозрачности имеют принципиальные ограничения, связанные с конечностью скорости передачи и обработки информации, поэтому необходимо свести к минимуму влияние на эти аспекты ошибок проектирования РИС.

Важным критерием при разработке РИС является определение степени прозрачности и поиск компромисса между прозрачностью и производительностью.

стью. Достижение прозрачности распределения - это разумная цель при проектировании и разработке распределенных систем, но она не должна рассматриваться в отрыве от других характеристик системы, таких как производительность, эффективность, стоимость и сложность.

Список литературы:

1. Интернет вещей, IoT, M2M мировой рынок [Электронный ресурс]. URL: [http://www.tadviser.ru/index.php/Статья:Интернет_вещей,_IoT,_M2M_\(мировой_рынок\)](http://www.tadviser.ru/index.php/Статья:Интернет_вещей,_IoT,_M2M_(мировой_рынок)) (дата обращения 27.11.2019).
2. ГОСТ Р ИСО/МЭК 10746-1-2004. Информационная технология (ИТ). Открытая распределенная обработка. Базовая модель. Часть 1. Основные положения. – Введ. 01.01.2005. – М. : Госстандарт России.
3. Макконел С. Совершенный код. Мастер-класс / Пер. с англ. – М. : Издательство: «Русская Редакция», 2010. – 896 с.
4. Блог компании Hewlett Packard Enterprise [Электронный ресурс]: Как добиться репликации с нулевым RPO на большие расстояния. URL: <https://habr.com/ru/company/hpe/blog/320366/> (дата обращения 25.01.2017).
5. URN: [Электронный ресурс]. URL: <https://ru.wikipedia.org/wiki/URN> (дата обращения 27.07.2019).
6. Э. Таненбаум Распределенные системы. Принципы и парадигмы / Э. Таненбаум, М. ван Стеен. – СПб.: Питер, 2010. – 880 с.: ил. - (Серия «Классика computer science»).
7. Péter Márton. Designing a Microservices Architecture for Failure // URL: <https://blog.risingstack.com> (дата обращения 15.08.2017).