

ИСТОРИЯ РАЗВИТИЯ ТЕХНОЛОГИЙ МОНИТОРИНГА ИНФОРМАЦИОННЫХ СИСТЕМ

Кореньков Владимир Васильевич¹, Ужинский Александр Владимирович²

¹ Кандидат физико-математических наук, профессор Института системного анализа и управления;

ГОУ ВПО «Международный Университет природы, общества и человека «Дубна»,
Институт системного анализа и управления;

141980, Московская обл., г. Дубна, ул. Университетская, 19;

Объединенный институт ядерных исследований, Лаборатория информационных технологий;

141980, Московская обл., г. Дубна, ул. Жолио-Кюри, 6;

e-mail: korenkov@lpub01.jinr.ru.

² Магистр техники и технологии по направлению «Системный анализ и управление», инженер-программист;

ГОУ ВПО «Международный Университет природы, общества и человека «Дубна»,
Институт системного анализа и управления;

141980, Московская обл., г. Дубна, ул. Университетская, 19;

Лаборатория информационных технологий, Объединенный институт ядерных исследований;

141980, Московская обл., г. Дубна, ул. Жолио-Кюри, 6;

e-mail: zalexandr@list.ru.

Любая система, стремящаяся к поддержанию своего состояния либо развитию, невозможна без механизмов контроля, которые в своей работе опираются на данные мониторинга. В работе рассматривается история развития технологий мониторинга информационных систем начиная с 60-х годов. Кратко изложены основные подходы и методы. Приведены наиболее интересные примеры реализации. Представлены различные техники мониторинга. Подробно изложены основы мониторинга распределенных информационных систем, компьютерных сетей, GRID, экологического мониторинга и т.д.

Ключевые слова: мониторинг, контроль, распределенные системы, экологический мониторинг, информационные системы, GRID.

HISTORY OF INFORMATION SYSTEM MONITORING EVALUATION

Korenkov Vladimir Vasilievich¹, Uzhinskiy Alexander Vladimirovich²

¹ PhD, professor of Institute of system analysis and management;

International university of the nature, society and man «Dubna»,
Institute of system analysis and management;

141980, Dubna, Moscow reg., Universitetskaya str., 19;

Join institute for nuclear research, Laboratory of information technologies;

141980, Moscow reg., Dubna, Joliot-Curie, 6;

e-mail: korenkov@lpub01.jinr.ru.

² Master of engineering and technology «Systems analysis and management», engineer-programmer;

International university of the nature, society and man «Dubna»,

Institute of system analysis and management;

141980, Dubna, Moscow reg., Universitetskaya str., 19;

Laboratory of information technologies, Join institute for nuclear research;

141980, Moscow reg., Dubna, Joliot-Curie, 6;

e-mail: zalexandr@list.ru.

Any system tended to maintain its state or evaluation requires control tools. The tools need data provided by monitoring system. History of monitoring technology evolution is considered in the paper starts from 60's. The main approaches and methods are shortly described. Most interesting examples and techniques are presented. Basic principles of distributed information system monitoring, computer networks, GRIDs, ecological monitoring, etc are given.

Key words: monitoring, control, distributed systems, ecology monitoring, information systems, GRID.

Введение

Мониторинг — это комплексный процесс, основная цель которого — получить представление о состоянии системы для принятия управленческих решений. Понятие «мониторинг» неразрывно связано с понятием «система». Практически любая система стремится к развитию и самосохранению, пытаясь избежать деградации и разрушения. Эти задачи решаются механизмом контроля, который получает информацию, необходимую для принятия решений от систем мониторинга. В большинстве систем (промышленных, социальных, экономических, биологических, информационных и т.д.) в том или ином виде имеются средства для мониторинга их состояния. В настоящей работе будет рассмотрена история развития лишь небольшого подмножества средств мониторинга, появившихся относительно недавно и используемых для получения информации о состоянии информационных систем.

Любая система, с точки зрения мониторинга, состоит из набора объектов, имеющих различные свойства (параметры). Значения данных параметров характеризуют состояния объектов, из которых складывается общее состояние системы. Показатели параметров фиксируются «наблюдателями» (сенсорами) и имеют такие характеристики, как актуальность и полнота. Перед создателями систем мониторинга всегда стоит ряд вопросов: для кого/чего создается система мониторинга, или, другими словами, кто будет пользователями системы; какие задачи должна решать система; какие объекты представляют наибольший интерес для пользователей; какие параметры объектов следует отслеживать; какие условия переходов от одного состояния в другое существуют в системе; каким образом и как часто в систему мониторинга будут поступать данные; в каком виде и как долго хранить собранные данные; в каком виде предоставлять данные пользователям?

Ответы на данные вопросы различались в зависимости от времени и класса информационных систем. В настоящей работе будут рассмотрены основные этапы развития систем мониторинга информационных систем, приведены некоторые концепции и методологии их создания, а так же наиболее интересные примеры из различных областей.

1. Вычислительные машины

В начале эры информационных систем, мониторинг был необходим как для автоматизации работы дорогих вычислительных установок, так и для обеспечения безопасности их работы. Системы были крайне уязвимы к ошибкам пользователей. Например, пользователи могли с легкостью уничтожить системную информацию или данные других пользователей, либо они могли продолжать «удерживать» за собой систему, даже после завершения своих вычислений, что фактически приводило к ее остановке. Для решения данных проблем были созданы достаточно простые средства разграничения прав доступа и контроля времени исполнения. Они позволили существенно сократить объем работы операторов вычислительных машин [40].

Уже в 60-х годах, наряду с системами механического мониторинга (набор температурных датчиков, внутренних счетчиков, таймеров и т.д.) [32], стали появляться программные системы мониторинга с графическим интерфейсом [17, 29]. Их основной задачей было отслеживание состояния

систем с разделенным временем. Подобные системы снимали показатели с определенной периодичностью, записывали их в хранилище данных, обрабатывали и предоставляли пользователям в доступном для понимания виде. Некоторые даже позволяли получать информацию в режиме on-line. Уже тогда разработчики систем мониторинга задавались вопросами: какие данные имеют большее значение; какие методы отображения информации наиболее доступны и понятны; какие значения показателей оптимальны для системы?

С развитием программного обеспечения и появлением не столь дорогих компьютеров, круг задач, решаемых информационными системами, значительно расширился. Компьютеры «доказали» свою самостоятельность и стали активно использоваться в различных областях, например, в медицине, для отслеживания состояния больных [12] или для решения различных задач в социальной сфере [31].

Кроме того, мониторинг стал незаменимым инструментом для улучшения производительности программного и аппаратного обеспечения [26, 3], а так же для уменьшения затрат на приобретение и поддержание вычислительных структур [8]. Сложность систем мониторинга возрастала, так же как и возложенная на них ответственность. Они использовались для снижения затрат, понимания внутреннего устройства вычислительных структур и определения будущих путей развития. Уже в то время задавались вопросы о полноте и непротиворечивости данных мониторинга, отслеживаемых метриках, а так же его «стоимости». Как упоминалось ранее, существовали системы как механического (аппаратного), так и программного мониторинга. Первые представляли собой набор сенсоров снимающих показатели с определенных объектов, стоимость таких систем в то время разнилась от 8 до 50 тысяч долларов. Программные системы представляли собой набор скрипов (небольших программ), приложений и хранилищ данных. Показания программных систем мониторинга могли расходиться с реальными данными, к тому же их использование создавало дополнительную нагрузку, в отдельных случаях до 40% системного времени. Однако, стоимость подобных систем была значительно ниже от 300 до 8 тысяч долларов. Проводились работы и в области создания гибридных систем, совмещающих в себе лучшие стороны вышеприведенных подходов [39].

В отдельную ветвь развития можно вынести системы мониторинга исполнения программ [1], которые кроме выполнения функций по контролю над ошибками (debugging), отслеживали изменение значения переменных, используемых в программах, используемую ими память и процессорное время, отображали ход исполнения задач и т.д.

2. Становление информационных систем

В семидесятых годах области применения различных информационных систем продолжали расширяться: бизнес, наука, социальная сфера, медицина и т.д. Использование данных систем порождало новые вопросы, на которые должны были отвечать системы мониторинга: оптимально ли работает система; что в системе надо поменять; когда надо это поменять; и на что это надо поменять? Так же определилась методология при создании систем мониторинга, включающая:

1. определение задач и цели мониторинга;
2. определение отслеживаемых параметров и характеристик, основываясь на поставленных целях;
3. проектирование и интегрирование структуры для мониторинга в систему;
4. проектирование и реализация средств анализа данных собранных системой мониторинга, если подходящие средства не доступны;
5. проектирование и инициирование экспериментальной ситуации, для получения данных;
6. выполнение анализа данных с последующими выводами, после завершения эксперимента;
7. определение путей улучшения системы, основываясь на данных мониторинга;
8. определение путей улучшения системы мониторинга. Данный пункт подразумевает увеличение/уменьшение числа отслеживаемых параметров, либо их качественное изменение, для получения более детализированной, либо более общей информации;
9. определение путей улучшения экспериментальной ситуации;

10. внесение необходимых изменений, определенных в пунктах с 7-го по 9-й;
11. после определенного времени, зависящего от поставленных задач и целей, повторение шагов с 5-го по 10-й.

В свою очередь, не оставались не освещенными и вопросы мониторинга систем с позиции системных администраторов. Мониторинг в данном аспекте должен был отвечать на вопросы: как используется система; какова производительность системы; каковы пути улучшения системы мониторинга? А также решать задачи предотвращения критических ситуаций и проектного анализа систем [13].

Довольно скрупулезно рассматривались вопросы анализа данных мониторинга: возможности применения различных статистических техник: структурного, корреляционного, регрессионного или факторного анализа, цепей Маркова и т.д. В качестве методов представления результатов предлагались текстовые отчеты, таблицы, а также различные диаграммы и графики.

3. 80-е годы

В конце 70-х годов активное развитие переживали реляционные базы данных, которые, впоследствии, внесли значительный вклад в развитие систем мониторинга. В то время они сами нуждались в мониторинге своего состояния, поскольку такие технологии, как сессии, транзакции, откаты и т.д. находились только в стадии разработки [7].

Еще одним несомненным достижением того времени стал механизм оповещения, который совместно с механизмом триггеров представлял удобный и надежный инструмент для управления базами данных. Суть механизма оповещения состояла в том, что пользователь мог определить набор различных правил, при нарушении которых ему бы выдавалось сообщение.

Для отображения результатов использовались новые, подчас довольно необычные техники. Так в работе [41] для отображения хода разработки программного обеспечения использовали графики с 8-ю осями - рис 1, по которым откладывался процент времени, потраченный при разработке программы на: 1 — отладку приложения; 2 — тестирование; 3 — проверку синтаксиса; 4 — написание кода; 5 — проектирование; 6 — управление проектом; 8 — на разработку подобного проекта ранее. На седьмой оси отображалось максимальное целочисленное значение произведения определенного параметра, служащего для масштабирования на количество операторов в программе.

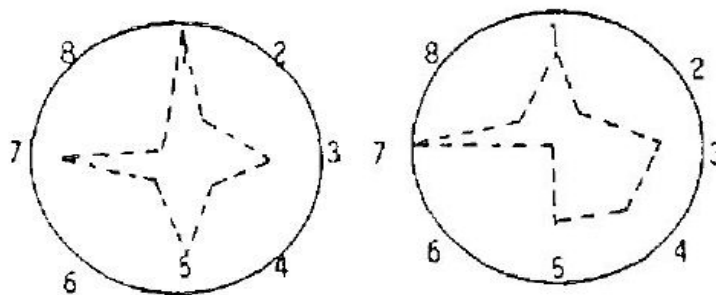


Рис. 1. Пример графического отчета

Еще одна популярная идея, оформившаяся в 80-х годах, заключалась в том, что компьютерные системы должны изначально разрабатываться с различными средствами мониторинга своего состояния. «Как и любые сложные системы, компьютерные системы должны быть оснащены набором средств помогающих при определении их состояния и потенциальных проблем» [39]. Ключевыми словами в данном высказывании являются «сложные системы». Как упоминалась во введении, любая система, стремящаяся к поддержанию своего состояния либо развитию, невозможна без механизмов контроля, которые в своей работе опираются на данные мониторинга. Таким образом, к 80-м годам вычислительные системы стали восприниматься именно в качестве сложных систем, обязанных поддерживать свое состояние на должном уровне.

Развитие объектно-ориентированного подхода в программировании, предложило еще один достаточно интересный механизм, решающий подобные задачи — «механизм проб». Пробы — экстренный запрос на предоставление информации о статусе, посылаемый определенным процессом другому процессу или объекту. Когда объект или процесс получает запрос на проведение пробы, он использует специальные процедуры, определенные по умолчанию, для формирования отчета о своем состоянии, а затем пересылает данный отчет приславшему запрос процессу. Процесс, отвечающий за снятие проб, выполняется постоянно с наивысшим приоритетом. Недоступные узлы системы могут быть с легкостью обнаружены, используя механизмы таймаутов (определенное время, отведенное узлу для отправки сообщения о своем состоянии) и повторных попыток. Кроме того, данные, полученные при проведении проб, могут быть использованы для восстановления состояния всей системы или ее отдельных объектов [10].

Таким образом, определились две основные техники снятия показателей системы. В первой сенсоры опрашиваются постоянно с определенной периодичностью, а во второй только при необходимости, например, после возникновения какого-либо события.

3. Распределенные системы

Распределенные системы — набор различных процессов работающих совместно для достижения определенной цели. Каждый процесс является обособленной программой, способной исполняться отдельно либо конкурируя с другими процессами. По ряду причин, отладка, тестирование и отслеживание изменений в распределенных системах являются более сложными задачами, нежели в централизованных.

1. Распределенные системы имеют много «точек контроля». Поэтому техники мониторинга и отладки централизованных приложений, такие как «tracing» или «breakpoints», должны быть «расширены» для применения в аспекте распределенных систем.
2. Из-за коммуникационных задержек при взаимодействии различных узлов распределенных систем, определение их состояния в конкретный момент времени является довольно сложной задачей. Например, инициализация запроса о состоянии системы производится с одного узла, а остальные узлы «узнают» о данном запросе только через некоторое, в общем случае неопределенное время.
3. Распределенные системы асинхронного времени по своей сути являются недетерминированными. Это значит, что запуск одинаковых заданий в подобных системах может привести к различным, но, тем не менее, правильным, последовательностям действий. Поэтому воссоздание в них ошибок и различных критических ситуаций является достаточно сложной задачей
4. Мониторинг распределенных систем изменяет их состояние, в то время как для централизованных систем подобных проблем не существует. Например, отладчик приложений может прервать последовательный процесс на «точке выхода», не затрагивая последующие процессы. В распределенных системах остановка или замедление одного из процессов может изменить состояние всей системы.
5. Взаимодействия между системой и разработчиком — предполагаемым пользователем средств мониторинга, могут быть достаточно сложными. Например, если терминал подсоединен к каждому процессору, то разработчику придется переходить от одного терминала к другому, для инициализации процессов, определения «точек выхода» и отслеживания последовательностей действий. Таким образом, необходимо обеспечить инструменты, облегчающие работу с системой и позволяющие контролировать процессы из единого центра [22].

Одним из требований к системе мониторинга распределенных систем является ее расширяемость, как физическая, так и функциональная. В основном, для этого применялся подход, в котором задачи определения и сбора данных отделялись от задач их обработки и предоставления пользователям.

Для представления данных использовались описанные ранее техники, с некоторыми уточнениями и улучшениями. Так, касательно текстовых отчетов, пользователям предоставлялась возможность использовать различные фильтры для уменьшения объема обрабатываемой информации; самостоятельно расставлять breakpoint'ы, так же предоставлялись дополнительные инструменты для работы с

историей исполнения событий. Для решения некоторых из описанных выше задач — улучшения читабельности текстовых отчетов, использования фильтров и определения условий передачи управления пользователю, применялись технологии паттернов — устойчивых составных частей. Например, запись «*: replay Calgary» может быть интерпретирована как «любой процесс отвечает Calgary», паттерном в данном случае служит конструкция {отправитель}: replay {получатель}. При работе с большими распределенными системами для пользователей очень важно иметь возможность сфокусироваться на процессах и событиях, которые в данный момент имеют наибольшее значение и возможность использования фильтров и breakpoint'ов позволяет решать эти задачи. Графические отчеты представляли собой набор символов, картинок и т.д. обозначающих состояние процессов и их взаимодействие друг с другом (рис 2).

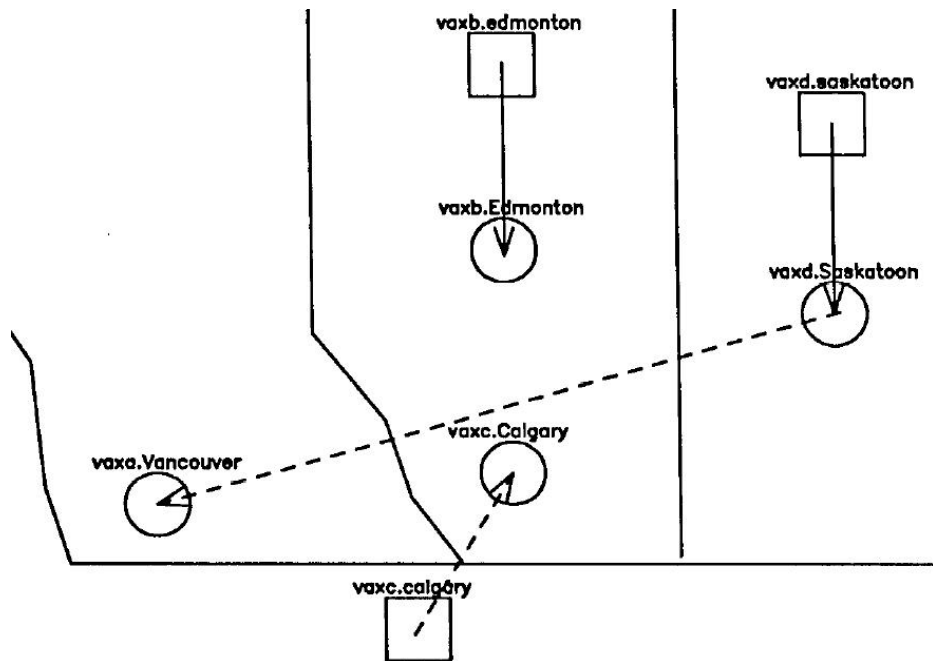


Рис 2. Пример графического интерфейса

У пользователей имелась возможность сохранять диаграммы и использовать мышь, для взаимодействия с графическими объектами, однако информация предоставляемая подобным образом была значительно менее полной, нежели предоставляемая текстовыми средствами. Применялись различные уровни абстракции — процессы могли рассматриваться как детально, так и в аспекте выполнения общей задачи, существовали механизмы увеличения или уменьшения детализации, объединения объектов в сообщества и т.д. Пользователи могли задавать шаг, с которым должна была перерисовываться рабочая область, поскольку события могли происходить настолько быстро, что отследить их становилось практически невозможно [33, 42].

Как тот, так и другой подход имели свои плюсы и минусы. Когда пользователь работал с текстовыми отчетами, все что видел пользователь, были появляющиеся строчки текста. Обычно они сменялись довольно быстро и понять, что же происходит в системе в целом, было достаточно сложно. В текстовых отчетах новая информация появляется в одном и том же месте, в то время как при работе с графикой, изменения могли произойти в любой части экрана. Это значит, что пользователю было непросто уловить эти изменения. С другой стороны, графические отчеты позволяли пользователю разносить различные объекты по разным областям экрана, что позволяло ему сконцентрироваться только на изменении интересующих его процессов. Таким образом, текстовые отчеты использовались в основном при отладке систем, а графические — для on-line мониторинга их состояния.

Мониторинг параллельных приложений — сходная с мониторингом распределенных систем область, которая так же переживала активное развитие в конце 80-х годов. Основным подходом в ней являлось использование «файлов событий», которые записывались во время исполнения программ, потом копировались на основной узел и анализировались [23]. Как и в распределенных системах в параллельных программах существовала проблема воспроизводимости сбоев. Решением данной проблемы являлось наличие в «файлах событий» всей необходимой информации для воссоздания

сбоя, однако подобный подход в свою очередь вел к значительному увеличению размеров файлов и к увеличению затрат на их формирование и обработку. Таким образом, разработчикам приходилось балансировать на тонкой грани между вышеприведенными крайностями.

Тем временем повальная компьютеризация общества стала общепризнанным фактом. Появлялись сотни систем, применявшихся для решения задач во множестве различных областей. Все эти системы нуждались в мониторинге. Космос [36, 34], робототехника [19], медицина [27], метеорология [34] — это только начало списка областей, в которых появлялись новые специализированные системы мониторинга.

Одной из наиболее интересных областей, в которой были собраны все новейшие технологии, была авиация. Системы мониторинга, использовались совместно с различными симуляторами и экспертными системами, для решения широкого круга задач: предоставление пилотам различной информации для принятия решений о последующих маневрах и траектории полета, управление автопилотируемыми средствами, разработка планов испытаний, отслеживание хода испытаний, моделирование различных ситуаций и т.д.

Общая схема работы подобных систем приведена на рисунке 3. Можно заметить, что основные вычисления производятся в центре на земле. При подобной организации изменения в программном или аппаратном обеспечении центра не приведут к необходимости замены оборудования в авиапарке.

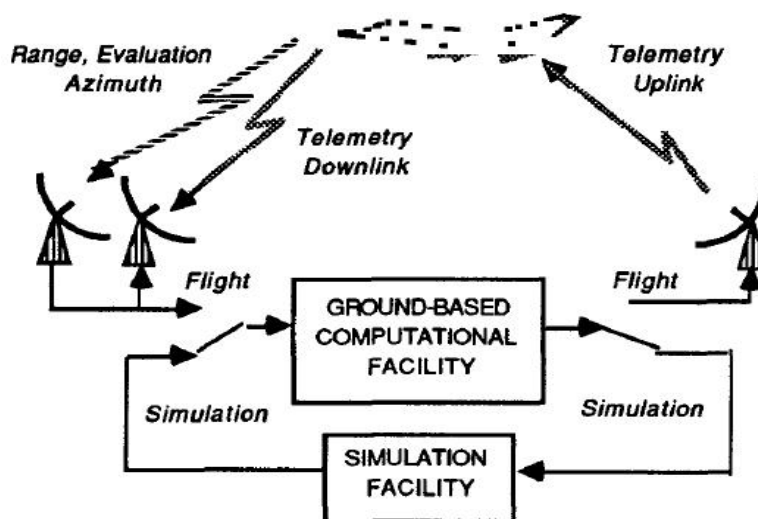


Рис. 3. Общая схема работы системы управления полетами

Естественно, интерфейсы взаимодействия между оборудованием, установленным на земле и на авиатехнике, должны были предоставлять максимум надежности при минимальном объеме передаваемой информации. Контролер траекторий, отвечающий за движение авиатехники, получал список маневров, которые следовало выполнить. Каждый маневр состоял из точки начала маневра, условий его выполнения и завершения. Каждый из данных пунктов мог иметь от трех до семи параметров. После получения инструкций, контролер траекторий производил необходимые действия без участия наземного центра, вплоть до получения дополнительных инструкций. В то же время на землю передавались все необходимые данные о прохождении полета.

Экспертные системы использовались при разработке плана полета, опираясь на базы знаний, они определяли очередность маневров, их приоритетность, а так же выстраивали траекторию полета. Каждый маневр детально моделировался и при возникновении каких-либо вопросов или осложнений оператор получал соответствующее сообщение. Приоритезация различных маневров была чрезвычайно важна при тестировании расходов топлива. Для удовлетворения условий заданных параметров, маневры с низким приоритетом могли быть исключены из плана полета.

Системы мониторинга опрашивали контролер траекторий для получения всех необходимых параметров полета. При недопустимом значении параметров, либо недостаточном качестве маневра, системы выдавали операторам соответствующие предупреждения и вырабатывали набор рекомендаций, основываясь на своих базах знаний, для решения проблемных ситуации.

Широкий спектр различных симуляторов, использовался как при работе экспертных систем, так и для тестовых испытаний без использования реальной авиатехники [20].

Со временем данное направление продолжало развиваться, вбирая в себя лучшее из смежных областей. Например, в 1995-ом году были проведены исследования на вопрос использования 3D дисплеев для мониторинга в реальном времени воздушного пространства. Результаты исследования показали, что скорость реакции на событие у операторов увеличилась на 23 процента, а количество ошибок уменьшилось на 60 [36]!

4. Компьютерные сети

С развитием всевозможных компьютерных сетей, развивались и методологии их мониторинга. Большинство систем создавались для удовлетворения запросов сразу нескольких групп пользователей. Руководители хотели знать, кто и как использует сетевые ресурсы. Сетевых инженеров интересовала информация о распределении и составляющих сетевого трафика. Операторы нуждались в данных для обнаружения и анализа сетевых аномалий. В самом общем случае, работа подобных систем сводилась к трем этапам: извлечение данных из заголовков пакетов, хранение и предварительная обработка данных, анализ данных [6].

4.1 Стандарты и спецификации для компьютерных сетей

Начиная разговор о компьютерных сетях вообще и их мониторинге в частности, невозможно не упомянуть так называемые RFC (Request for Comments) документы. Появившийся еще в 1969 году формат документов, в 80х-90х годах стал общепризнанным стандартом для компьютерных сетей и Интернета. Существует огромное количество различных стандартов, и рассмотрение даже небольшой части из них, это тема для отдельного исследования.

В аспекте мониторинга, стандарты описывали свойства, состояния, методы, а так же протоколы взаимодействия между объектами сети (хостами, свитчами, маршрутизаторами, роутерами и т.д.). Основным плюсом подобного подхода являлось то, что не зависимо от аппаратного и программного обеспечения при использовании стандартных команд и запросов, должен быть получен один из стандартных ответов.

Вот лишь небольшой список документов, используемых при мониторинге сетей: RFC0777: Internet Control Message Protocol (1981) — протокол управления сетью, используется для определения доступности узлов, анализа различных показателей и т.д.; RFC0869: Host Monitoring Protocol (1983) — протокол для сбора информации об узлах в различных сетях; RFC1022: High-level Entity Management Protocol (HEMP) (1987) — протокол для управления различными объектами сети; RFC1067: Simple Network Management Protocol (1988) — протокол управления сетями, используется для получения информации с узлов сети (маршрутизаторов, свичей и т.д.); RFC1156: Management Information Base for network management of TCP/IP-based internets (1990) — протокол и основные структуры для управления сетями; RFC1147: FYI on a Network Management Tool Catalog: Tools for Monitoring and Debugging TCP/IP Internets and Interconnected Devices (1990) — описание различных средств мониторинга и управления сетями.

Создавались различные приложения как отдельные, так и интегрированные в операционные системы, позволяющие использовать данные протоколы для получения информации об объектах или изменения их состояния. Таким образом, на нижнем уровне находились стандартизированные протоколы и примитивные средства для работы с ними, а на верхнем — различные приложения, обрабатывающие полученную на нижнем уровне информацию и предоставляющие ее в доступном виде.

4.2 Развитие мониторинга компьютерных сетей

Компьютерные сети возникали повсеместно, и одной из основных проблем была нехватка, а так же высокая стоимость услуг специалистов в данной области. Это послужило толчком к созданию различных экспертных систем для анализа данных мониторинга. В своей работе подобные системы опирались на данные о протоколах (размеры пакетов, таймауты, контрольные суммы и т.д.), тополо-

гии сетей, версиях используемых приложений и т.д. В большинстве случаев, для анализа использовались данные различных уровней. Самый нижний из уровней содержал информацию об отдельных пакетах, либо суммарную информацию о пакетах. На следующем уровне существовали так называемые «bunches» — наборы пакетов, не всегда упорядоченные и «landmarks» — наборы пакетов, относящиеся к определенному узлу. Далее наборы пакетов собирались в последовательности, а они в свою очередь, в подключения. Информация о подключениях предоставляла данные о трафике, а полная информация обо всем трафике давала представление об общем состоянии сети. Опираясь на приведенную выше информацию и оперируя правилами или наборами правил, экспертные системы могли решать довольно широкий спектр задач, значительно облегчая труд сетевых администраторов. [21, 4].

Компьютерные сети продолжали развиваться, и из небольших Ethernet сегментов перерастали в огромные мульти-сегментарные структуры, с разделяемой Backbone и некоторым количеством подключенных посредством роутеров.

Было понятно, что обрабатывать все без исключения пакеты и отслеживать все подключения становилось слишком накладно, а подчас практически невозможно. Системы мониторинга начали отделять «интересные» подключения от «неинтересных». Информация о «интересных» подключениях, такая как данные о пакетах, интервалы между пакетами, объемы передаваемой информации и т.д. в полном объеме хранилась и обрабатывалась программами, в то время как информация о «неинтересных» отбрасывалась. Существовали так же механизмы хранения информации о потенциально интересных подключениях, которые могли быть со временем перенесены в «интересные», либо стертые, а также механизмы переоценки «интересных» подключений, которые могли со временем быть перенесены в «неинтересные». В основном, для определения «интересности» использовались такие характеристики как общее число пакетов и частота их появления [24]. Одним словом обработка каждого пакета в отдельности изжила себя, и каждый пакет стал рассматриваться в аспекте его принадлежности к определенному подключению.

Впоследствии значительную часть работы по обнаружению и предотвращению сбоев и сетевых атак взяло на себя коммуникационное оборудование, однако роль сетевых администраторов в функционировании сетей остается довольно значительной.

Естественным дополнением к возможности получать информацию о работе сети стали различные приложения, позволяющие на основе собранного материала моделировать работу сети, для определения «узких мест», возможностей для оптимизации, возможных вариантов развития и т.д. Описание одной из подобных систем, совмещающей в себе средства для мониторинга и моделирования можно найти в [43].

5. Мониторинг с человеческим лицом

В конце 80-х стало понятно, что мониторинг — это комплексный процесс, который должен эволюционировать. Восприятие пользователями системы со временем меняется, и уровень их удовлетворенности системой сильно зависит от опыта работы с ней. Качество системы, это комбинация ее производительности и степени в которой она отвечает запросам пользователей в сфере информированности о ее состоянии. Подключение потенциальных пользователей на этапе проектирования и реализации системы позволяет удовлетворить большинство их потребностей [18]. Каждый разработчик должен помнить эти простые и понятные правила.

Здесь будет не лишним упомянуть и об экологическом мониторинге. Число исчезающих видов животных увеличивается с каждым годом, отслеживать все изменения в данной области старыми методами было просто невозможно, вследствие чего появился новый подход, при котором отслеживалось состояние всей экосистемы в целом. Для определения данного состояния использовались наиболее характеризующие ее параметры (различные группы растений, животных и т.д.). Используя данные со спутников, передающих информацию об изменениях на земной поверхности, появилась возможность автоматически отслеживать подобные характеристики. Спутники передавали в наземные центры огромное количество графической информации — терабайты данных в день. Для анализа этих данных использовались многопроцессорные вычислительные структуры и программы, применяющие различные параллельные алгоритмы.

Чтобы понять общее состояние экосистемы, нужно знать схемы взаимодействия всех ее компонентов: геологических, климатических, вегетативных, биологических и т.д. Ключевые параметры должны быть отобраны по принципам точности показателей и минимизации затрат на их получение. Например, птицы входят в подмножество наиболее интересных кандидатов, поскольку наблюдать за ними значительно легче, нежели за животными, рептилиями или насекомыми. Птицы проводят значительную часть своей жизни на открытых пространствах (в полете), их сезонные миграции происходят практически в одно и то же время и проходят по одним и тем же маршрутам. Естественно, что флора и топография местности тоже хорошо поддаются анализу и предоставляют огромное количество полезной информации.

Как было замечено ранее, объемы данных огромны, вследствие чего их обработка производилась в три этапа. На первом этапе, использовались довольно «недорогие» алгоритмы и большие срезы данных, определялись кандидаты в области возможного обитания интересующих исследователей видов. Затем в работу включались алгоритмы высокого уровня, более затратные, однако достаточно точные, чтобы из всего списка кандидатов выбрать наиболее подходящих. На втором этапе данные о выбранных областях переводились в удобный для обработки вид, поскольку 2D графика является далеко не идеальным для обработки форматом. На третьем этапе к областям кандидатам применялись высокоточные алгоритмы (зачастую параллельные) для определения интересующих исследователей параметров [42, 15]. Таким образом, работа операторов по определению исследуемых характеристик сводилась к минимуму, и они могли спокойно заниматься анализом.

Бурное развитие в конце 20-го века технологий беспроводных сетей, предоставило широчайшие возможности для создания различных систем мониторинга, в том числе и для мониторинга биосферы. Различные портативные устройства (сенсоры) теперь могут быть установлены где угодно и снимать практически любую информацию, а современные технологии позволяют создать миниатюрные передатчики, которые могут быть закреплены на исследуемых образцах животных, не причиняя им особого неудобства. Данные, фиксируемые сенсорами, передаются в командные центры для хранения и обработки, после чего становятся доступны для исследователей в удобном для работы виде [26].

Скорее всего, подобный подход к мониторингу будет активно развиваться и укреплять свои позиции.

6. Новое тысячелетие

В 21-ом веке широчайшее применение получило целое семейство технологий, зародившиеся еще в конце 20-го века. Объем информации по ним огромен (по каждой написан не один десяток книг), и даже рассмотрение общих концепций привел бы к увеличению объема настоящей работы в несколько раз, вследствие чего, они будут упомянуты лишь вскользь.

Огромные корпорации, предприятия и сообщества стремились максимально автоматизировать и оптимизировать протекающие в них бизнес-, технологические-, коммуникационные- и прочие процессы. Для решения подобных задач применялись таких технологий как ERP (Enterprise Resource Planning System) — корпоративная информационная система, предназначенная для автоматизации учета и управления, CRM (Customer Relationship Management) — управление взаимоотношениями с клиентами, HRM (Human Resource Management) — управление персоналом, PLM (Product Lifecycle Management) — управление жизненным циклом изделий, MRP (Material Requirement Planning) — управление производством и запасами и другие. Учитывая, что одной из основных задач подобных технологий является обеспечение надежных механизмов контроля и управления, мониторинг играет в них важнейшую роль.

Новые концепции требовали новых технологий хранения и анализа данных, и они не заставили себя ждать. В основном, все они являются развитием технологии называемой Business intelligence (BI), разработанной для сбора, объединения, анализа и представления различной информации о бизнес процессах. Вот лишь некоторые из них: Data mining — процесс обработки огромных массивов данных с целью получения значимой информации, DWH (Data Warehouse) — хранилища данных, спроектированные специально для облегчения анализа данных, OLAP (Online Analytical Processing) — технологии для максимально удобной работы с комплексными многомерными данными.

Следует заметить, что стоимость серьезных продуктов, использующих данные технологии, довольно внушительна, что делает возможность их применения для средних, малых, научных, социальных и прочих организаций, не обладающих солидными бюджетами, довольно затруднительной. Однако их потенциальные возможности огромны и можно смело утверждать, что именно они находятся на передовой в области сбора, хранения и анализа данных.

7. GRID

GRID — еще одна концепция получившая широкое распространение в 21-ом веке. Ее основной принцип — объединение разнородных и территориально распределенных ресурсов в единую высокопроизводительную вычислительную структуру. Центральным элементом GRID является middleware (промежуточное программное обеспечение), основными строительными блоками которого являются grid-сервисы, необходимые для решения различных задач и выполненные в сервисной архитектуре (Service Oriented Architecture) на базе Web-сервисов, удовлетворяющих спецификации WS-I (Web Services Interoperability). Теоретические основы данной концепции см. в [14].

Системы мониторинга GRID призваны решать довольно широкий спектр задач: обнаружение новых ресурсов, определение лучшего узла для расчета задачи или файла для скачивания, выявление проблем, анализ производительности, оптимизация элементов и т.д. Хотя GRID — относительно новое понятие, в его основе многое опирается на компьютерные сети и распределенные вычисления. Соответственно различные модификации техник и методологий мониторинга данных областей успешно применяются в GRID.

В аспекте GRID мониторинга распространены такие понятия как «information provider» — обеспечивает доступ к информации об узлах сети и «aggregate directory service» — отвечает за сбор и обработку информации от «information provider», зачастую так же выполняет различные управленческие функции. Многие используемые протоколы являются расширением протоколов для обычных сетей. В GRID прослеживается четкая линия создания средств мониторинга. Разработчики middleware создают специальные сервисы, отвечающие за сбор и распространение данных об элементах сети: Globus — MDS (monitoring and discovery service) [9, 44], European DataGrid — R-GMA (Relational-Grid Monitoring Architecture) [11], Condor — Hawkeye [45], а за анализ и предоставление информации пользователям обычно отвечают другие средства, разрабатываемые отдельными группами в рамках одного или нескольких экспериментов. Существуют примеры создания комплексных средств мониторинга, так же основанных на технологиях Web-сервисов, получающие информацию из различных источников и в отдельных случаях позволяющие расширять список получаемых параметров работы grid — MonALISA [46], GridICE [2], BOSS [16], MapCenter [5] и т. д.

Таким образом, задачи сбора и обработки информации отделяются от ее анализа и предоставления. Однако стоит заметить, что зачастую возможности анализа довольно ограничены. Технологии экспертных систем и систем поддержки принятия решений используются крайне редко. Адаптивности и самоорганизации уделяется очень мало внимания, что в свете тенденций последних лет, значительно влияет на интерес к GRID в целом.

Большинство приведенных проблем связано с тем, что сама технология GRID находится только на этапе становления. Скорее всего, как это было и с компьютерными сетями, стандартизация и спецификации предоставят возможности для применения подходов упомянутых выше, что самым благоприятным образом скажется на удобстве, качестве и надежности GRID-сервисов.

Заключение

В последнее время одной из довольно популярных идей является самоорганизация вычислительных систем. Компьютерные структуры настолько огромны, что управлять ими «вручную» сложно и довольно накладно, вследствие чего появился новый подход, в рамках которого на приложения перекладываются практически все обязанности администраторов. Подобные системы должны сами производить операции установки и настройки в гетерогенной среде, оптимизации для достижения максимальной производительности, защиты от атак и сбоев, а также обязаны заниматься самолечением, т.е. определять сбои, диагностировать неработающие компоненты и производить комплекс меро-

приятый для исправления ситуации [27, 29]. Естественно, что функционирование подобных систем не возможно без эффективного мониторинга.

Вычислительные системы все больше и больше входят в нашу жизнь. Хорошо это или плохо — вопрос неоднозначный. Понятно одно, это данность, с которой следует считаться и сделать все, чтобы системы, которые не только облегчают человеку жизнь, но зачастую и несут за нее ответственность, были максимально надежны. Кроме того, вычислительные системы приобретают все большую схожесть со сложными биологическими или социальными системами, приобретая способность адаптироваться к условиям окружающей среды, развиваться и бороться с возникающими проблемами. Механизмы контроля играют в данных процессах решающую роль, при этом они не возможны без надежных и удобных средств мониторинга. В свете последних рассуждений, мониторинг является и будет являться одной из приоритетных составляющих в развитии вычислительных систем.

Список литературы

1. Aygun, B. Environments for monitoring and dynamic analysis of execution / B. Aygun // *Simulation of Computer Networks*, 1973. — P. 178-197.
2. Aftimiei, C. Recent evolutions of GridICE: a monitoring tool for grid systems / C. Aftimiei, S. Andreozzi, G. Cuscela, G. Donvito, V. Dudhalkar, S. Fantineli, E. Fattibene, G. Maggi, G. Misurelli, A. Pierro // *Proceedings of the 2007 workshop on Grid monitoring*, 2007. — P. 1-8.
3. Arndt, F. Hardware Monitoring of Real-Time Computer System Performance / F. Arndt, C. Oliver // *Computer*, 1972. — vol. 5, issue 4. — P. 25-29.
4. Barton, B. A real-time expert system for computer network monitor and control / B. Barton, J. Switlik // *ACM SIGMIS Database*, 1988. — vol 19, issue 2. — P. 35-38.
5. Bonnassieux, F. Mapcenter: an open grid status visualization tool / F. Bonnassieux, R. Harakaly, P. Primet // *In Proceedings of the 15th International Conference on Parallel and Distributed Computing Systems*, 2002.
6. Braden, R. A pseudo-machine for packet monitoring and statistics / R. Braden // *SIGCOMM Computer Communication Review*, August 1988. — vol 18, issue 4. — P. 200-209.
7. Buneman, O. Efficiently monitoring relational databases / O. Buneman, E. Clemons // *ACM Transactions on Database Systems (TODS)*, September 1979. — vol. 4, issue 3. — P. 368-382.
8. Carlson, G. How to save money with computer monitoring / G. Carlson // *ACM Annual Conference / Annual Meeting*, 1972. — P. 1018-1023.
9. Czajkowski, K. Grid Information Services for Distributed Resource Sharing, In Proc / K. Czajkowski, S. Fitzgerald, I. Foster, C. Kesselman // *10th IEEE International Sym.p. on High Performance Distributed Computing (HPDC-10)*, IEEE Press, 2001.
10. Dasgupta, P. A probe-based monitoring scheme for an object-oriented distributed operating system / P. Dasgupta // *Conference on Object Oriented Programming Systems Languages and Applications*, 1986. — P. 57-68.
11. DataGrid Information and Monitoring Services Architecture: Design, Requirements and Evaluation Criteria // *Technical Report*, DataGrid, 2002.
12. Deller, S. Utilization of a small computer for real-time continuous patient monitoring / S. Deller, F. Lewis, M. Quinn // *ACM Annual Conference/Annual Meeting*, 1971. — P. 622-639.
13. Dominick, W. Automated monitoring to support the analysis and evaluation of information systems / W. Dominick, W. Penniman // *ACM SIGIR Forum*, September 1979. — vol. 14, issue 2. — P. 2-9.
14. Foster, I. The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration / C. Kesselman, J. Nick, S. Tuecke // *tech. report*, Glous Project January, 2002.
15. Frances, M. Analysis of multi-interpretable ecological monitoring information / M. Frances, T. Brazier, J. Engelfriet, J. Treue // *Lecture Notes In Computer Science*, January 1998. — vol. 1455. — P. 303-324.

16. Grandi, C. Object Based System for Batch Job Submission and Monitoring (BOSS) / C. Grandi, A. Renzi // CMS NOTE-2003/005.
17. Grochow, J. The graph display as an aid in the monitoring of a time shared computer system / J. Grochow // Technical Report: TR-54, 1968.
18. Gynes, C. Monitoring database performance - a control issue / C. Gynes // SIGSAC Review, March 1987. — vol. 5, issue 2. — P. 7-11.
19. Hardy, N. Declarative sensor knowledge in a robot monitoring system / N. Hardy, D. Barnes, M. Lee // Proc. of the NATO Advanced Research Workshop on Languages for sensor-based control in robotics, 1987. — P. 169-187.
20. Hewett, M. The development of an automated flight test management system for flight test planning and monitoring / M. Hewett, D. Tartt, E. Duke, R. Antoniewicz, A. Agarwal // International conference on Industrial and engineering applications of artificial intelligence and expert systems, 1988. — P. 324-333.
21. Hitson, B. Knowledge-based monitoring and control: an approach to understanding behavior of TCP/IP network protocols / B. Hitson // ACM SIGCOMM Computer Communication Review, Volume 18, Issue 4, August 1988. — P. 210-221.
22. Joyce, J. Monitoring distributed systems / J. Joyce, G. Lomow, K. Slind, B. Unger // ACM Transactions on Computer Systems (TOCS), Volume 5, Issue 2, May 1987. —P. 121-150.
23. Kerola, W. MONIT: a performance monitoring tool for parallel and pseudo-parallel programs / W. Kerola, H. Schwetman // SIGMETRICS Performance Evaluation Review, Volume 15, Issue 1, May 1987. — P. 163-174.
24. Lorence, M. IPWATCH: a tool for monitoring network locality / M. Lorence, M. Satyanarayanan // ACM SIGOPS Operating Systems Review, Volume 24, Issue 1, January 1990. — P. 58-80.
25. Lucas, H. Performance Evaluation and Monitoring / H. Lucas // ACM Computing Surveys (CSUR), Volume 3, Issue 3, September 1971. —P. 79-91.
26. Mainwarding, A. An analysis of a large scale habitat monitoring application / A. Mainwarding, D. Culler, J. Polastre, R. Szewczyk, J. Anderson // Conference On Embedded Networked Sensor Systems, 2004. — P. 214-226.
27. Mandell, A. Dynamical complexity and pathological order in the cardiac monitoring problem / A. Mandell // Physica D, 1987. — vol. 27, issue 1-2. — P. 235-242.
28. Munawar, M. Monitoring multi-tier clustered systems with invariant metric relationships / M. Munawar, M. Jiang, P. Ward // Proceedings of the 2008 international workshop on Software engineering for adaptive and self-managing systems, 2008. — P. 73-80.
29. Pikerton, T. Performance monitoring in a time-sharing system / T. Pikerton // Communications of the ACM, November 1969. — vol. 12, issue 11. — P. 608-610.
30. Reidemeister, T. Model-based Symptoms for Problem Determination in Enterprise Software Systems / T. Reidemeister, M. Munawar, P. Ward // Proceedings of the of the Fourth International Workshop on Engineering Autonomic Software Systems, 2007.
31. Roman, R. Monitoring and glurp in the ISVD and beyond / R. Roman // ACM SIGCUE Outlook June 1970. — vol. 4, issue 3. — P. 2-11.
32. Schulman, F. Hardware measurement device for IBM system/360 time sharing evaluation / F. Schulman // ACM Annual Conference/Annual Meeting, 1967. — P. 103-109.
33. Schwan, K. Real-time monitoring of parallel and distributed systems / K. Schwan, D. Ogle // The Ohio State University, 1988. — P. 200.
34. Clair, D. ESPAD: an adaptively controlled rule-based expert system for monitoring and diagnosing space vehicle subsystems / D. Clair, A. Wetterstroem, V. Johnson // ACM Annual Computer Science Conference, 1988. — P. 667-672.

35. Stain, D. On-line documentation and tutorials for an integrated graphical and data analysis system / D. Stain // ACM SIGCHI Bulletin, Volume 19, Issue 2, October 1987. — P. 56-57.
36. Stark, G. Monitoring software reliability in the shuttle mission simulator / G. Stark // Proceedings of the 1987 Fall Joint Computer Conference on Exploring technology: today and tomorrow, 1987. —P. 123-124.
37. Steinberg, D. 3-D displays for real-time monitoring of air traffic / D. Steinberg, C. Deplachett, K. Pathak, D. Strickland // Conference on Human Factors in Computing Systems, 1995.
38. Svobodova, L. Online system performance measurements with software and hybrid monitors / L. Svobodova // ACM Symposium on Operating Systems Principles, 1973. — P. 45-53.
39. Svobodova, L. Performance monitoring in computer systems: a structured approach / L. Svobodova // SIGOPS Operating Systems Review, Volume 15, Issue 3, July 1981. — P. 39-50.
40. Swift, C. Machine Features for a More Automatic Monitoring System on Digital Computers / C. Swift // ACM (JACM), April 1957. — vol. 4, issue 2. — P. 172-173.
41. Tanik, M. Software development monitoring graphs / M. Tanik // SIGSOFT Software Engineering Notes, April 1980. — vol. 5, issue 2. — P. 34-37.
42. Tokuda, H. Real-time monitor for a distributed real-time operating system / H. Tokuda, M. Kotera, C. Mercer // Workshop on Parallel & Distributed Debugging, 1988. — P. 68-77.
43. Turner, C. Adaptive data parallel methods for ecosystem monitoring / C. Turner, J. Turner // Conference on High Performance Networking and Computing, 1994. — P. 281-290.
44. Winters, G. An architecture for monitoring and modeling network systems / G. Winters, Z. Zhu, M. Bauer, H. Lutfiyya, D. Muntz, T. Teorey // IBM Centre for Advanced Studies Conference, 1995.
45. Официальный сайт MDS [Электронный ресурс]. — Режим доступа: <http://www.globus.org/toolkit/mds/> (дата обращения: 04.06.2009).
46. Официальный сайт проекта Hawkeye [Электронный ресурс]. — Режим доступа: <http://www.cs.wisc.edu/condor/hawkeye> (дата обращения: 04.06.2009).
47. Официальный сайт системы MONALIS [Электронный ресурс]. — Режим доступа: <http://monalisa.cacr.caltech.edu/monalisa.htm> (дата обращения: 04.06.2009).