

Министерство образования и науки Российской Федерации
Сибирский федеральный университет

Р. Ю. Царёв
А. В. Прокопенко
А. Ю. Никифоров

ОСНОВЫ РАСПРЕДЕЛЕННОЙ ОБРАБОТКИ ИНФОРМАЦИИ

Рекомендовано УМО РАЕ по классическому университетскому и техническому образованию в качестве учебного пособия для студентов высших учебных заведений, обучающихся по направлениям подготовки: 09.03.03 «Прикладная информатика», 09.03.04 «Программная инженерия», 38.03.02 «Менеджмент», 38.03.05 «Бизнес-информатика» (рег. № 528 от 10.08.2015 г.)

Красноярск
СФУ
2015

УДК 004.032.2(07)

ББК 32.971.3я73

Ц181

Р е ц е н з е н т ы:

А. Н. Антамошкин, доктор технических наук, профессор Красноярского государственного аграрного университета;

А. А. Ступина, доктор технических наук, профессор Красноярского государственного аграрного университета

Царёв, Р. Ю.

Ц181 Основы распределенной обработки информации : учеб. пособие / Р. Ю. Царёв, А. В. Прокопенко, А. Ю. Никифоров. – Красноярск : Сиб. федер. ун-т, 2015. – 180 с.
ISBN 978-5-7638-3386-7

Рассмотрены основы распределенной обработки информации и проблемы, связанные с проектированием и разработкой распределенных систем. Детально описаны процессы параллельной обработки данных и показаны современные подходы к решению возникающих при параллельной обработке информации задач. Особое внимание уделено вопросам распределенной обработки информации и управления при решении производственных задач. Представлены модели оценки программного обеспечения распределенных систем.

Предназначено для студентов, обучающихся по направлениям 09.03.03 «Прикладная информатика», 09.03.04 «Программная инженерия», 38.03.02 «Менеджмент», 38.03.05 «Бизнес-информатика», всех форм обучения.

Электронный вариант издания см.:
<http://catalog.sfu-kras.ru>

УДК 004.032.2(07)
ББК 32.971.3я73

ISBN 978-5-7638-3386-7

© Сибирский федеральный
университет, 2015

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	5
1. АВТОМАТИЗАЦИЯ МЕТОДОЛОГИЙ МОДЕЛИРОВАНИЯ И УПРАВЛЕНИЯ РАСПРЕДЕЛЕННЫМИ СИСТЕМАМИ	8
1.1. Анализ задач распределенной обработки информации.....	8
1.1.1. Задачи управления ресурсами	9
1.1.2. Интегрированная методология планирования ресурсов	10
1.1.3. Задачи корпоративного управления.....	11
1.1.4. Интегрированные решения	12
1.1.5. Проблемы информационной поддержки методологии логистических цепочек.....	13
1.2. Системный анализ задач распределенной системы	15
1.2.1. Анализ системотехнических задач управления	15
1.2.2. Системотехнический анализ распределенной системы	20
1.3. Распределенное управление и обработка информации	25
1.4. Структурные методологии и анализ	31
1.4.1. Концепция структурного анализа	31
1.4.2. Классификация структурных методологий	36
1.4.3. Примеры структурных методологий	40
2. МОДЕЛЬНЫЕ КОМПОНЕНТЫ СТОХАСТИЧЕСКОЙ СТРУКТУРЫ АЛГОРИТМОВ РАСПРЕДЕЛЕННОЙ ОБРАБОТКИ ИНФОРМАЦИИ.....	59
2.1. Стохастическое представление моделей формирования	59
2.1.1. <i>GERT</i> -сетевая модель стохастической структуры	59
2.1.2. Минимизация затрат ресурсов	62
2.1.3. Случайные акции при реализации алгоритмов распределенной обработки информации	64
2.1.4. Многократное исполнение задач распределенной обработки информации	67
2.1.5. Минимизация по времени.....	68
2.2. Стохастическая модель определения нормативных времен распределенной обработки информации в условиях неопределенности.....	70
2.2.1. <i>GERT</i> -сетевое представление моделей процессов	70
2.2.2. Примеры применения системы <i>GERT</i> -моделей	81

2.2.3. Определение вероятностных нормативных времен для процессов, реализуемых в условиях неопределенности	84
2.2.4. Пример реализации <i>GERT</i> -анализа в системах технического обслуживания тягового подвижного состава	90
2.3. Сетевые модели анализа и тестирования программного обеспечения распределенных систем	108
2.3.1. Сетевое представление и автоматизация подготовки алгоритмов с использованием сетей Петри	110
2.3.2. Анализ модели программного обеспечения распределенной обработки информации	119
3. РАСПРЕДЕЛЕННЫЕ СИСТЕМЫ ОБРАБОТКИ ИНФОРМАЦИИ И УПРАВЛЕНИЯ НА ПРОИЗВОДСТВЕ.....	125
3.1. Средства автоматизированного системотехнического проектирования распределенных систем.....	125
3.2. Структура и средства информационной поддержки компонент распределенной системы.....	134
3.2.1. Концептуальная архитектура и состав системы программной поддержки многокомпонентной модели	135
3.2.2. Функционирование системы программно-алгоритмической поддержки	138
4. УПРАВЛЕНИЕ ПРОЕКТАМИ ПРИ РАСПРЕДЕЛЕННОЙ ОБРАБОТКЕ ИНФОРМАЦИИ	142
4.1. Метрики измерения программного обеспечения распределенной системы и процесса его разработки	146
4.1.1. Размерно-ориентированные метрики	146
4.1.2. Функционально-ориентированные метрики	147
4.1.3. Выполнение оценки проекта на основе <i>LOC</i> - и <i>FP</i> -метрик	154
4.2. Конструктивная модель расчета стоимости программного обеспечения распределенной системы	156
4.2.1. Модель композиции приложения	157
4.2.2. Модель раннего этапа проектирования	159
4.2.3. Модель этапа постархитектуры	163
4.3. Предварительная оценка программного обеспечения.....	166
4.4. Анализ чувствительности программного обеспечения	170
ЗАКЛЮЧЕНИЕ	176
БИБЛИОГРАФИЧЕСКИЙ СПИСОК	178

ВВЕДЕНИЕ

Современный подход к созданию распределенных систем состоит в объединении в единую систему или сеть множества обрабатывающих средств (процессоров), средств управления, хранения и обработки информации (разноплатформенные СУБД, различные учетные системы), средств обмена и коммутации структуры [10].

На этапе системотехнического проектирования распределенных систем одной из важнейших является задача формирования алгоритмов распределенной обработки информации. На заданной структуре аппаратно-программных средств необходимо осуществить выбор системных и прикладных программ, структур данных и способов взаимодействия этих компонентов, обеспечивающих заданный ресурсно-временной режим реализации информационно-алгоритмических задач распределенной системы.

При создании моделей и алгоритмов распределенной обработки информации необходимо учитывать, что в распределенных системах режим реального времени предполагает лимитирование времени ответа системы управления на запрос объекта [9]. Ограничение на время реакции связывается в этом случае с выполнением периодических действий. При этом, начиная с момента первоначального запроса, все будущие моменты запроса периодической задачи можно определить заранее путем прибавления к моменту начального запроса величины, кратной известному периоду. Таким образом, при реализации периодичных задач формирование алгоритмов распределенной обработки информации должно осуществляться с учетом ограничений, представленных в форме классов ресурсов, жесткого регламента задач и временных пределов реализации задач.

Естественной математической интерпретацией распределенных, асинхронных и мультипрограммных систем являются сетевые модели, которые позволяют отражать распределенность структуры, сетевой характер взаимосвязей между процессами и ресурсами, а также между аппаратными и программными компонентами распределенных систем. В связи с этим для решения задач системного анализа и формирования алгоритмов распределенной обработки информации привлекается сетевой анализ, для реализации которого используется многокомпонентная сетевая модель.

Общность методов построения управляющих проектирования и реализации распределенных систем, выражающаяся в использовании комплекса сетевых моделей, позволяет в рамках многокомпонентной сетевой

модели создать единые средства автоматизированного формирования алгоритмов распределенной обработки информации.

В первой главе проведен анализ современных методологий моделирования и управления распределенными системами. Отмечается, что именно предпринятые сегодня расширения классических методик управления ресурсами распределенных систем дают толчок для новых витков развития методологий управления системами и технологий системного моделирования.

Проанализированы основные направления развития методологии *MRP*, часть из которых выделилась в самостоятельные методологии автоматизированного управления, например, управление сложными проектами, аналогичными разработкам на заказ, где планирование ведется по совмещенным сетевым графикам. Проведен анализ системотехнических задач, решение которых необходимо обеспечить при проектировании распределенных систем и формировании алгоритмов распределенной обработки информации.

Во второй главе в качестве формальной базы алгоритмической *GERT*-процедуры используется аппарат стохастических сетей и графического метода оценки и пересмотра планов. Рассмотрены модели временной реализации распределенных алгоритмов, которые описываются детерминированными структурами *GERT*-сетей. Приведены процедуры для этапа анализа реализуемости и коррекции распределенных алгоритмов с учетом алгоритмически заданных ограничений, которые соответствуют общей задаче различимости и включают исследование совместных свойств алгоритмов распределенной обработки информации. Также показан способ представления моделей программ, реализующих алгоритмы, в виде сетей Петри и предложен набор элементов модели, которые позволяют описывать базовые абстракции и механизмы реализации алгоритмов распределенной обработки информации.

Третья глава посвящена описанию формального аппарата многокомпонентной сетевой модели формирования алгоритмов распределенной обработки информации в виде программной системы, которая предоставляет пользователю (специалисту проблемной области) удобные средства для решения специфических задач в составе инструментальной системы.

Система программно-алгоритмической поддержки многокомпонентной модели формирования распределенной обработки информации реализована в виде модуля «Многокомпонентная модель» *ERP*-системы *MBS-Axapta*. Тот факт, что язык описания моделей алгоритмов распределенной обработки информации приближен к табличной форме, позволяет использовать большие объемы числовых данных, реализованные на основе реляционных баз данных и технологий «клиент-сервер». Это обеспечивает эффектив-

ность применения высокоуровневых методов информатики и программирования и использования на их основе таких преимуществ баз данных, как независимость от прикладных программ; минимальную избыточность данных, так как одними и теми же данными можно пользоваться в различных компонентах модели при решении различных задач и т. д.

В четвертой главе освещаются вопросы планирования проекта при распределенной обработке информации и оценки затрат проекта, обсуждаются размерно-ориентированные и функционально-ориентированные метрики затрат, методика их применения. Достаточно подробно описывается наиболее популярная современная модель оценки затрат – *COCOMO II*. В качестве иллюстраций приводятся примеры предварительного оценивания проекта, анализа влияния на проект конкретных условий разработки.

1. АВТОМАТИЗАЦИЯ МЕТОДОЛОГИЙ МОДЕЛИРОВАНИЯ И УПРАВЛЕНИЯ РАСПРЕДЕЛЕННЫМИ СИСТЕМАМИ

1.1. Анализ задач распределенной обработки информации

Анализ современных методологий автоматизированного управления распределенными системами показывает, что предпринятые сегодня расширения классических методик управления ресурсами дают толчок для новых витков развития методологий управления системами и технологий системного моделирования. Однако на практике математическая поддержка процесса моделирования в рамках современных методологий используется явно недостаточно, что отнюдь не всегда обосновано.

Касаясь технологий системного моделирования, можно отметить, что наиболее широкое применение нашли статистические методы. Однако формальное основание для применения указанных методов существует далеко не всегда. Часто используются предположения о независимости, одинаковой распределенности и нормальности изучаемых совокупностей случайных величин. Существуют ситуации, в которых данные предположения могут не выполняться, и это обстоятельство может приводить к потере точности и достоверности результатов моделирования.

Важнейшей задачей моделирования является проверка адекватности модели, которая, вообще говоря, является предметом рассмотрения фундаментальных наук. В идеальной ситуации, когда известны законы, справедливые для изучаемой модели, заключения об адекватности могут быть сделаны по результатам проверки этих законов в процессе моделирования.

Для сложных систем, к классу которых относятся и распределенные системы, возможность проведения такой проверки предоставляется редко. Поэтому обычно выводы об адекватности модели делают на основе проверок соответствий машинной и исходной математической моделей, поведений моделируемой и реальной систем, а также правильности интерпретации результатов моделирования.

Соответствие модели и системы, прежде всего, гарантируется структурным подобием. Проверка адекватности модели в указанном смысле осуществляется путем привлечения эвристик, доступной информации

о моделируемой системе, применения статистических методов, анализа чувствительности модели по отношению к ее параметрам.

1.1.1. Задачи управления ресурсами

Задача управления ресурсами в распределенных системах возникла практически одновременно с методом Тейлора – Форда. Рассмотрим применение распределенных систем и управление ресурсами на производстве.

С математической точки зрения проблема расчета потребностей для штучного производства не представляла особых трудностей и легко решалась интуитивно-ручными методами. Но при резком увеличении количества товаров и особенно при их замене или модификациях проблемы резко возрастали. Неудивительно, что первые массовые производства испытывали значительные трудности при смене моделей. Для решения задачи распределенного управления была разработана методология планирования материальных ресурсов – *MRP (Material Requirements Planning)*. Однако оказалось, что кроме методических трудностей здесь имеются и математические, которые полностью могут быть решены, только используя современные распределенные системы.

Использование этой методологии подразумевает, как правило, применение *MPS «Master planning schedule»*, хорошо известной под названием «объемно-календарное планирование», методологии, которая является базовой практически для всех планово-ориентированных методологий. Достаточно быстро был реализован вариант планирования ресурсов – *CRP «Capacity Requirements Planning»*.

Эта методология во многом похожа на *MRP*, но обеспечивает расчеты требуемых ресурсов, а не только материалов и компонентов. Эта задача существенно сложнее, поскольку требует учета большого числа параметров, а окончательный расчет обязательно включает не только объем ресурсов, но и временную последовательность.

Стандартная задача расчета ресурсов «знает» об ограниченных мощностях обрабатывающих центров распределенной системы, но максимум, что она может сделать – это рассчитать «потребность» во времени обработки информации для выполнения запланированной программы при неограниченном горизонте планирования либо показать превышения (недостаток) потребных мощностей при ограниченном горизонте. Если результат оказывается неудовлетворительным, то требуется изменить программу и повторить процесс. Поскольку это весьма ресурсоемкая вычислительная задача, которая даже на современных распределенных системах выполняется часами, то очевидна ее математическая *нетехнологичность*.

1.1.2. Интегрированная методология планирования ресурсов

Объединение рассмотренных выше методологий обеспечивает решение задачи *MRP* «второго уровня» – *MRP II* «*Manufacturing Resource Planning*». Ее реализация возможна только на базе интегрированной методологии планирования, включающей *MRP/CRP*. При использовании данной методологии обязательно подразумевается анализ финансовых результатов плана, а также применение *MPS* и *FRP* «*Finance Resource/Requirements Planning*» – планирование финансовых ресурсов (как правило, без их интеграции в «динамическую систему»).

С учетом недостаточной мощности компьютеров были разработаны методологии чернового планирования ресурсов – *RCCP*, которые позволили формировать график без проведения полной процедуры расчета, а затем уже производить окончательный баланс ресурсов по обеим «ветвям» планирования – как по *MRP*, так и по *CRP*. Именно на этом уровне данная задача предлагается на сегодняшний день в виде тиражируемых решений, а именно систем класса *MRP II*.

Отметим, что в таком виде задача планирования ресурсов представляет интерес только для ограниченного числа «типичных *MRP* (*MRP II*) – производств»: машиностроение; приборостроение; серийные сборочные производства.

Для указанных типов задача расчета потребных ресурсов является *самоценной* ввиду своей вычислительной сложности. При этом модель – одна, а реализаций может быть много.

Для большинства случаев «расчет чистых потребностей» оказался недостаточным, что привело к дальнейшему развитию «постановок» задач. Выделяют несколько основных направлений развития методологии *MRP*, часть из которых позже выделилась в самостоятельные методологии управления:

- управление сложными проектами, аналогичными разработкам на заказ, где планирование ведется по совмещенным сетевым графикам (такое «проектное управление» используется в тяжелом машиностроении, авиастроении, космической отрасли и др.);
- интегрированное управление для заказного и мелкосерийного производства (машиностроение, автомобилестроение и др.);
- управление сложными структурами, включающими холдинговое управление, финансово-промышленные группы, крупные торгово-производственные компании.

Каждая из перечисленных задач имеет специфические требования к функциональности программного обеспечения (ПО) распределенных систем. Например, «финансовое управление» требует значительно более мощного механизма аналитического учета и бюджетного управления, чем

это необходимо для производственного предприятия, а «управление распределенными потребностями» – специального механизма планирования и организации межскладских перемещений, не связанного непосредственно с планированием потребностей (в том числе, с технологической точки зрения, – поддержки механизма репликации, автономных транзакций и/или глобальных сетей).

1.1.3. Задачи корпоративного управления

Два последних направления лежат в основе методологий управления так называемыми компаниями типа *FMCG* (*fast moving consumer good* – быстродвижущиеся потребительские товары). Самостоятельность этих направлений означает возможность их первоначальной реализации в рамках отдельной системы и важность такой реализации для бизнеса компании. Естественно на каком-то шаге становится очевидно, что неинтегрированная задача дает только частичные или временные преимущества и требуется строить интегрированную систему.

Остановимся на методах формализации постановки задачи корпоративного управления. Опыт постановки задач для реальных распределенных систем и анализ представленных в России реализаций распределенных систем позволили выделить три подхода к формальной постановке задач: функциональный, финансовый и «документооборотный».

Чистых решений, основанных на одном из указанных подходов, не существует, однако преимущества или недостатки каждого подхода очень сильно сказываются на возможностях и качественных характеристиках систем.

В настоящее время очевидны преимущества *функционального подхода* в области управления, типичными представителями которого являются решения от *Baan* и *Symix*. Финансовый подход проповедует компания *SAP* в своем продукте *R/3*. Это оказалось очень эффективно для управления предприятиями холдингового типа и для управления чисто финансовыми институтами.

Исключительно популярный в России *документооборотный* вариант среди западных систем в чистом виде, по-видимому, не встречается вовсе, хотя его влияние сильно ощущается, например, в *Oracle Application*. Отмечается, что в существенной степени этот вариант встречается только в малых системах и в непромышленной сфере, где мал удельный вес встроенных вычислительных задач, либо они легко отчуждаются в самостоятельные модули. Важно отметить, что те или иные реализации документооборота все больше включаются в состав многих систем автоматизированного управления, однако их информационные задачи достаточно четко очерчены как внешние по отношению к основной функциональности системы. Также распространенным является вариант интеграции систем

финансово-экономического управления и системы, обеспечивающей реализацию того или иного варианта документооборота (в том числе таким, как *Lotus Notes* или *Microsoft Outlook*).

Таким образом, тиражируемые *MRP*-системы, как правило, дополненные хотя бы элементарными системами управления складами и финансами, начали свое победное шествие в начале 80-х (заказные и уникальные появились значительно раньше). Но и сейчас их низкая функциональность приводит к необходимости существенных доработок и, следовательно, существенных затрат на внедрение и так называемую адаптацию продукта, а фактически – разработку недостающей функциональности. Это обосновывается тем, что на смену стандартным требованиям поддержки информационно-ориентированной методологии управления пришли требования поддержки информационно-алгоритмических методологий.

1.1.4. Интегрированные решения

Наибольшие проблемы возникают в случае потребности в реализации интегрированного решения хотя бы в размере стандартной функциональности «средней» системы типа *Symix SyteLine* или *Ross Renaissance*.

В настоящее время внедрение автоматизированной системы «замораживает» ранее внедренные управленческие решения, реализованные часто еще в период мэйнфреймов. Кроме того, в 1980-х сформировались новые характеристики рынка, которые существенно изменили требования к программному обеспечению распределенных систем. Потребовалась переработка концепций, заложенных в основу автоматизации производства. Возникла потребность в интегрированных решениях, охватывающих и связывающих планирование и управление всеми сферами деятельности, включая производственные мощности, материальные (товарные) и финансовые ресурсы. Данный подход был подхвачен практически всеми основными поставщиками *MRP II* систем. Реально, однако, не предлагалось ничего нового, а лишь происходила констатация достигнутого уровня постановки задач для тиражируемых решений больших систем.

Концепция управления ресурсами – *CSRP (Customer Synchronized Resource Planning)* – «планирование ресурсов, синхронизированное с потребителем», была предложена компанией *Symix*. Сущность данной концепции состоит в том, что при планировании и управлении можно и нужно учитывать не только основные ресурсы, но и все те ресурсы, которые обычно рассматриваются как вспомогательные или накладные. Это все ресурсы, потребляемые во время маркетинговой и текущей работы с клиентом, послепродажного обслуживания проданных товаров, перевалочных и обслуживающих операций, а также внутрицеховых ресурсов. Это позволяет учитывать все этапы жизненного цикла товара.

Косвенным, но исключительно важным следствием данной концепции явилась реализация – впервые в системах класса *ERP* – задачи тонкого управления производственными графиками в условиях ограниченных мощностей (так называемой *APS*-задачи – расширенного управления графиками).

Автономные решения такого класса были известны и раньше, но впервые была достигнута интегрированность с полноценным *ERP*-пакетом. Системы типа *APS* позволяют решать такие задачи, как «проталкивание» срочной заявки в графики, распределение заданий с учетом приоритетов и ограничений, перепланирование с использованием полноценного графического интерфейса. Благодаря принципиально новой «математике» расчет типовых *MRP*-задач происходит на несколько порядков быстрее, нежели раньше. Итак, стало реальностью ежечасное изменение графика, что в условиях «классической» *ERP*-задачи относилось к категории *нерешаемых* задач. Детальный анализ трудоемкости заявки стал возможен уже на этапе его оформления, причем с учетом конкретных технологических решений.

1.1.5. Проблемы информационной поддержки методологии логистических цепочек

Важнейшей глобальной новинкой в области управления стала концепция логистических цепочек (*supply chain*). Информационная поддержка данной методологии – обязательное требование для крупных предприятий. Характерно, что еще в 2001 году *R/3*, которая позиционируется как система управления, прежде всего, для крупных систем, активно критиковалась именно за недостаточный уровень поддержки данной концепции.

Переход в практике и теории управления от манипулирования поставками (*supply management*) к управлению цепочками поставок (*supply chain management*) или, точнее, управлению логистическими цепочками произошел совсем недавно – в течение последнего десятилетия.

Сущность понятия «логистическая цепочка» можно раскрыть на основе примера учета при анализе производства всей цепочки (точнее сети), по которой товар из сырья превращается в готовое изделие и затем через систему продаж попадает к конечному потребителю.

Например, понятие «управление продажами» включает в себя одностороннее рассмотрение события, происходящего только на последнем этапе логистической цепочки. Причем это очень короткий участок цепочки – самое большое «продавец – потребитель», а рассмотрение обычно ведется только с точки зрения выталкивания товара на рынок. Но даже здесь применение расширенной концепции логистических цепочек (анализ функционирования системы продаж во всей ее сложности, включая предыдущую и последующую стадии продажи) позволяет поднять анализ происходящего на новый уровень.

Возникновение теории и практики управления логистическими цепочками связано с прогрессом информационных технологий, который позволил проводить операции и анализ деятельности в режиме *on-line*. Естественно, это потребовало осмысления и формализации методологий управления и разработки соответствующих инструментов. Поддержка логических цепочек с 2002 года стала практически обязательным требованием для программных продуктов, предназначенных для распределенных систем.

Именно такие продукты должны поддерживать конфигурацию, позволяющую размещать объекты автоматизации на нескольких физически удаленных территориях, со всеми вытекающими отсюда требованиями к распределенной структуре базы данных. Во многих случаях также необходим вариант «тонкого» клиента для обеспечения рабочих мест, например, для дистанционного формирования заявки или мониторинга.

Здесь следует отметить концепцию *DRP*, которая реализована достаточно давно, например, в таких продуктах, как *CA-PRMS*, а также в заказных системах. Особенность системы в том, что она вполне качественно работает с *off-line* информацией. Нет принципиальных сложностей при ее реализации с помощью *Excel*. Проанализировав сущность анализа и формирования логистических цепочек, можно отметить, что введение понятия «логистическая цепочка» было не менее революционным, нежели переход в управлении к концепции *MRP II*.

К сожалению, термин *Supply Chain* не может считаться точно формализованным. Реализация данной концепции в программных продуктах различна, поэтому при выборе решения необходимо тщательно знакомиться с конкретной функциональной реализацией.

В заключение хотелось бы обратить внимание на один аспект. Методологии *Supply Chain* и *CSRP* взаимно дополняют друг друга. Первая фокусируется на глобальной логистике и связанных с ней внешних процессах. Вторая – на внутренних процессах, в частности, на тонком управлении заявками и расширенном управлении издержками благодаря трактовке бизнес-процессов как «расширенного» цикла и, что важно, не «товара вообще», как *MRP*, а «товара в конкретном производственном заказе», что точно соответствует идеологии *Supply Chain*.

Учитывая, что «ядром» логистических цепочек является производитель, можно сказать, что методология *CSRP* – это методология производственного ядра *Supply Chain*. Объединение этих двух методологий в единой системе позволило бы выйти на новый качественный уровень распределенных систем и методологий управления ресурсами. Автоматизированные системы, поддерживающие тонкое аналитическое управление распределенными системами и логистическими цепочками (на информационно-алгоритмическом уровне), могут дать очень значительные преимущества.

Исходя из общих тенденций развития программного обеспечения для распределенных систем можно предположить, что такого рода подходы будут предложены и начнут реализовываться в ближайшие годы.

1.2. Системный анализ задач распределенной системы

Создание распределенных программно-информационных технологий базируется на объединении в единую систему или сеть множества обрабатывающих средств (процессоров), средств хранения, обработки и управления информацией (разноплатформенные СУБД, различные учетные системы), средств обмена и коммутации структуры.

На этапе системотехнического проектирования одной из важных задач является задача формирования алгоритмов распределенной обработки информации и управления. На заданной структуре распределенных аппаратно-программных средств необходимо осуществить выбор системных и прикладных программ, структур данных и способов взаимодействия этих компонентов, обеспечивающих заданный режим применения программно-информационных технологий.

1.2.1. Анализ системотехнических задач управления

Определим основные задачи, решение которых необходимо или целесообразно обеспечить при проектировании распределенной системы и формировании алгоритмов распределенной обработки информации и управления в системе.

1. Гибкость распределенной системы обеспечивается разработкой управляющих моделей, представленных структурами данных, а также их интерпретацией с помощью единой операционной системы моделирования и управления. Математическое обеспечение программного управления строится, как правило, путем реализации совокупности базовых управляющих моделей, адаптируемых к конкретным технологическим процессам и условиям на основе распределенной обработки информации и управления.

2. Взаимодействие с объектом по жестким временным диаграммам и режим реального времени. Управление с помощью распределенной системы предполагает выполнение функций управления локальными модулями, координацию их действий с действиями различных подсистем. Взаимодействие с объектом обычно осуществляется по принципу прямого цифрового управления, реализуемого средствами с числовым программным управлением. Запуск следующей операции может выполняться по сигналам от службы времени управляющего процессора в соответствии с последова-

тельностью директивных сроков управления. В этом случае организуется действие на объект или взаимодействие с ним по жестким временным диаграммам.

Строгое регламентирование диаграммы взаимодействия управляющей системы и объекта возможно лишь при монопольной однопрограммной организации работы микропроцессора. Во взаимодействии по жесткой временной диаграмме режим реального времени присутствует в наиболее строгой своей форме. Более простая форма режима реального времени предполагает лимитирование времени ответа распределенной управляющей вычислительной системы на запрос от объекта. Ограничение на время реакции связывается в этом случае с выполнением периодических действий. В мультипрограммных системах реальный масштаб времени задается предельным временем пребывания заявок в системе, включающим время ожидания в очередях и обслуживания процессорами и другими устройствами.

3. Корректность системных действий и взаимодействий. Переход от централизованных к распределенным системам породил высокий параллелизм и асинхронность информационно-алгоритмических процессов в распределенных системах. Взаимодействующие процессы могут изменять свое состояние под управлением средств операционной системы или при поступлении необходимых данных. Эффективность распределенной системы во многом зависит от способов синхронизации между процессорными элементами, работающими совместно над множеством задач.

Асинхронный принцип управления не предписывает заранее, в какой последовательности должны активизироваться процессы. Последовательность определяется в ходе вычислений и формирования управлений. Это обеспечивает некоторую подстройку динамики управляющей системы к реальному ходу управляемых процессов. Источниками асинхронности могут быть, например, непредсказуемые задержки в действиях управляемого оборудования, в коммутационных линиях, при использовании процессоров.

В распределенных системах это сопровождается перестройкой алгоритмического и программного обеспечения, порядка использования ресурсов, протоколов локального взаимодействия процессов. В связи с такими качественными изменениями могут стать возможными нарушения системного взаимодействия как в пределах локальной управляющей вычислительной системы, так и в пределах автоматизированной системы в целом.

Проверка качественной корректности системных действий предполагает анализ следующих свойств процессов: живость, отсутствие тупиков, ограниченность, безопасность. Процесс (или оператор) называют *живым*, если существуют выполнимые условия его активизации. *Тупик* – это сис-

темная ситуация, связанная с взаимной блокировкой нескольких процессов, захвативших коммунальные системные ресурсы. В распределенных системах ресурсы выделяются несколькими контроллерами по запросам от процессов. Каждый контроллер в состоянии избежать локальных тупиков, но в силу параллелизма взаимодействующих процессов может возникнуть *распределенный тупик*. Предсказание тупика на этапе системного проектирования должно исключить тупиковую остановку процессов управления. Высокая ответственность распределенной системы должна обеспечиваться также средствами обнаружения тупиков, возможность возникновения которых не была предсказана на этапе анализа корректности системных взаимодействий. Отрицательные последствия возникшего и обнаруженного тупика должны быть ликвидированы с минимальными потерями. Специалист по технологии управления должен исключить возможность повторения такой тупиковой ситуации.

Термин *ограниченность* связывают с ограниченной емкостью системных очередей, буферов данных и т. п. Если в системе невозможны ситуации, приводящие к переполнению системных очередей, буферов данных и других накопителей, то говорят, что условие ограниченности для системных действий выполнено. Нарушения ограниченности могут привести к потере сообщений или данных.

Термин *безопасность* связывают с устранением критических состояний распределенной системы. Безопасность двух процессов распределенной обработки информации и управления нарушается, если алгоритмам одновременно удастся получить доступ к ресурсу, предназначенному для персонального использования другого алгоритма обработки данных или управления объектом.

4. Распараллеливание алгоритмов. Задача состоит в представлении алгоритмов распределенной обработки информации и управления в такой форме, которая обеспечивает быстрое и эффективное использование информации о взаимной зависимости операторов. Многогранность данной проблемы связана с разнообразием архитектур современных вычислительных систем, языков представления алгоритмов, способов преобразования алгоритмов из исходной формы в заданную.

5. Распределение работ между распределенными ресурсами. Распределение вычислительных работ между обрабатывающими средствами вычислительной системы или сети выполняется с целью обеспечения функций управления в режиме реального времени; повышения производительности обработки информации; эффективного использования распределенных процессорных мощностей распределенной системы. Планы распределения работ создают на основе использования информации о возможном параллелизме в выполнении операторов алгоритма. Задача распределения работ

состоит в наилучшем приспособлении алгоритмов к заданной архитектуре распределенной системы и представляет собой *задачу синтеза распределенного алгоритма при заданной структуре (архитектуре) системы*.

Поставленная задача может формулироваться также в виде задания на подбор смеси работ для данной сети процессоров и других ресурсов. Смеси подбирают на основе известного числа обращений к ресурсам, взаимного временного расположения и времен обслуживания запросов. Если к смешиваемым работам предъявляют различные требования по оперативности их обработки, то подбор смесей сопровождается выбором типа квантования или приоритетного обслуживания и распределением приоритетов по работам.

6. Комплексирование. Эта задача состоит в приспособлении архитектуры распределенной системы к целесообразной организации функций распределенного управления. Данная задача является обратной к сформулированной выше задаче распределения работ. По отношению к техническим средствам распределенной системы можно выделить два типа операций комплексирования: компоновка распределенной системы – тактическая операция, связанная с существенной реорганизацией ее физической структуры при изменениях алгоритмов распределенной обработки и управления; редактирование структуры распределенной системы – оперативное действие, выполняемое в пределах имеющегося комплекса технических средств, как правило, автоматически.

Задания на компоновку и редактирование формулируют путем подбора логической структуры системы, которую можно построить в пределах заданной физической структуры распределенной системы, обеспечивая эффективное ведение процессов обработки информации и управления. Математически данная задача ставится как задача синтеза структуры, наилучшим образом приспособленной к выполнению набора алгоритмов распределенной обработки информации и управления, при заданных ограничениях на возможность комплексирования.

7. Оптимизация СУБД. Одной из критических задач проектирования распределенных систем является задача оптимизации быстродействия системы управления базами данных (СУБД), так как это существенно влияет на повышение доступности информационных ресурсов, вплоть до обеспечения гарантированной доступности.

Рассматривая СУБД как компонент программно-информационных технологий, следует отметить, что это сложный и высоко настраиваемый программный продукт. Его гибкость позволяет оказывать влияние на информационную производительность среды в целом, при этом требуются лишь небольшие настроечные изменения, которые, тем не менее, произвести «вручную» практически невозможно. Однако, как показывает опыт