**Лекция № 15 (32)**– 08.12.23 г. БФСМ (окончание)

### **8.2.8. Динамическая базовая функционально-структурная модель**

### **архитектуры АСОИУ**

Множество проектных задач ℳ(𝓏), с учетом сопровождающих его динамических явлений на периоде проектирования системы, представим формально в следующем виде:

= ∮(ℳ (𝓏), τ), (9.4)

ℳ(𝓏) – множество проектных задач, построенное, в общем случае, на основе коллегиально - иерархического графа Γ(κι),

τ– время, τ =; Τ – период проектирования системы,

∮ - оператор преобразования,

- функционально-структурная модель архитектуры АСОИУ.

Рассмотрим оператор преобразования ∮, который по существу раскрывает физический смысл динамической функционально-структурной модели архитектуры АСОИУ. Для этого применим аналитический аппарат теории графодинамики, основы которой приведены в Приложении Б.

Воспользуемся формальным графом интегральных свойств АИС, подобным графу, который представлен на рис. 9.15-б предыдущей лекции. Каждая пронумерованная вершина этого графа должно строго соответствовать единственной и определённой вершине исходного вербального графа, аналогичного графу на рис. 9.15-а. Это соответствие должно сохраняться при любых преобразованиях формального графа интегральных показателей свойств АИС. Поскольку этот граф задаёт множество проектных задач, то назовём его графом проектных задач или, проще, **графом задач**.

Применим для описания графа задач функцию подчинения (П - функцию) φ(n), n = , N–число пронумерованных вершин графа.

Ранее был рассмотрен жизненный цикл простой системы и его графическая модель (рис. 3.8) в виде последовательности сменяющих друг друга стадий, определяющих конкретное состояние системы, а именно:

исследование реализуемости системы,

технико-экономическое обоснование,

формирование технического задания,

разработка технического предложения,

эскизное проектирование,

техническое проектирование,

рабочее проектирование,

испытания,

опытная эксплуатация,

промышленная эксплуатация,

модернизация системы с продолжением её эксплуатации,

утилизация системы.

Каждая стадия характеризуется своим из множества ℳ (𝓏) составом проектных задач, по которым возможно принятие конкретного проектного решения в объеме тех знаний о системе, которые соответствуют стадии.

Каждой стадии жизненного цикла поставим в соответствие граф задач, описываемый соответствующей П-функцией. Тогда последовательность сменяющих друг друга графов, то есть графовая траектория, может быть представлена как последовательность П- функций

φτ=0(n) → φτ=1(n) → φτ=2(n) → , , , → φτ=T(n), (9.5)

где T – число учитываемых стадий жизненного цикла,

подчиняясь рекуррентному соотношению

φτ+1(n) = F[φτ(n)], (9.6)

где F – оператор преобразования.

В общем случае в (9.5) φ(n) может быть деревом или лесом. При развёртывании этой последовательности во времени многосвязный граф (многие корни и растущие из них деревья) постепенно переходит в односвязный (дерево), внутри данного дерева могут возникать переподчинения вершин и т.п. Более того, в роли аргумента (независимых переменных) в (9.6) могут выступать графы, принадлежащие графовой траектории в предыдущие моменты времени (соответствующие предыдущим стадиям жизненного цикла), так что будет иметь место уравнение графодинамики вида

φτ+1(n) = F[φτ(n), φτ-1(n), …, φτ+1(n)]. (9.7)

Наконец, в роли независимых переменных можно рассматривать, например, граф задач φ1(n1) и оргграф φ2(n2), изменения которых во времени сами задаются рекуррентными соотношениями, и тогда для описания графодинамических процессов проектирования системы возникают уравнения вида

φτ+1(n) = F[φτ(n),φτ-1(n),… ,φ1τ+1(n1) = Fφ1(φ1τ(n1), φ1τ-1(n1), … ),

φ2τ+1(n2) = Fφ2(φ2τ(n2), φ2τ-1(n2)]. (9.8)

В процессе проектирования графовая траектория приходит в равновесное состояние такое, что

φ∗(n) = F[φ∗(n)], (9.9)

где φ∗(n) – граф, отвечающий законченному проекту создаваемой системы (или соответствующий периоду эксплуатации системы).

Процесс установления равновесия должен закончиться за конечное число шагов, так как на проектирование отводится ограниченное время.

На рис. 9.16 приведена графодинамическая модель жизненного цикла сложной системы, являющаяся конкретизацией рассмотренной ранее графической модели (рис.3.9). При создании автоматизированной информационной системы, как правило, должны учитываться существующие на объекте фрагменты автоматизации, бесперспективные в контексте будущей системы, которые, во-первых, работоспособны и привычны для пользователей и, во-вторых, по капиталоёмкости не

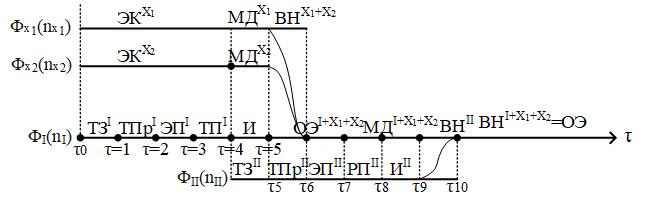


Рис. 9.16. Графодинамическая модель жизненного цикла

сложной системы

Обозначения: x1, x1 – наследственные фрагменты создаваемой системы,

I, II – первая и вторая очереди создаваемой системы,

ЭКx1, ЭКx2– периоды эксплуатации наследственных фрагментов,

МДx1, МДx2– стадии модернизации наследственных фрагментов (подготовки к слиянию с созданной очередью системы),

ВНx1+x2– стадия внедрения (слияния фрагментов x1 и x2 с первой очередью),

ТЗ, ТПр, ЭП, ТП, РП, И, ОЭ – стадия жизненного цикла соответственно технического задания, технического предложения, эскизного проектирования, рабочего проектирования, испытаний, опытной эксплуатации,

φx1(nx1), φx2(nx2), φI(nI), φII(nII) – графовые траектории соответственно фрагментов x1 и x2, первой I и второй II очереди системы,

τ – время

рациональны для утилизации. Благоразумно использовать их в составе новой системы с учётом того, что в дальнейшем они могут быть эволюционно модернизированы, вплоть до полного исключения из состава системы.

Эти фрагменты представлены на рис. 9.16, как наследованные фрагменты. Кроме того, представленная здесь модель соответствует системе, создаваемой двумя очередями: первой-I и второй – II. Стадии МДх1 и МДх2 предполагают техническую подготовку наследованных фрагментов к слиянию с I-ой очередью системы, осуществляемому на стадии ВНх1+х2.

Таким образом, в графодинамической модели жизненного цикла сложной системы присутствуют ветви, характеризующие развитие во времени наследованных элементов Х1 и Х2, которые действовали до создания системы, а также первой Iи второй II очередей создаваемой системы. В рассматриваемом жизненном цикле с момента времени τ = 6 совместно функционируют I-я очередь и фрагменты Х1 и Х2, с τ = 10 они функционируют совместно со II-ой очередью.

Жизненный цикл на рис.9.16 можно описать уравнением

φτ+1(n) = Fφ[φτ(n), φτ-1(n), … ,φх1τ+1(nх1) = Fφх1(φх1τ(nх1), φх1τ-1(nх1), … ),

φх2τ+1(nх2) = Fφх2(φх2τ(nх2), φх2τ-1(nх2), … ),φIτ+1(n1) = FφI(φIτ(nI),

φIτ-1(nI), … ), φIIτ+1(nII) = FφII(φI{τ(nII), φIIτ-1(nII), … )]. (9.10)

Начиная с τ = 10 графовая траектория (9.10) приходит в равновесное состояние такое, что

φ∗(n) = F[φ∗(n)], (9.11)

где φ∗(n) – граф, отвечающий действующей системе в составе I-ой и II-ой очередей и фрагментов X1 и X2. Процесс установления равновесия должен закончиться за конечное число шагов, так как на создание системы отводится ограниченное время.

Физический смысл уравнения (9.10) состоит в том, что оно описывает полное множество согласованных между собой проектных задач и его динамику, обусловленную необходимостью развития системы. Моменты нововведений определяются привходящими условиями (высвобождением ресурсов для расширения разработки, появлением на рынке более совершенного оборудования, новые требования пользователей системы и т.п.). Однако возникнув, они диктуют необходимость формулировки новых проектных задач в контексте множества уже реализованных задач, заданных уравнением (9.10). Удовлетворение системным условиям является фактором необходимым для того, чтобы очередное нововведение не понизило эффективность функционирования системы ниже требуемой величины.

Применение аппарата графодинамики приводит в рассматриваемом прикладном аспекте к необходимости решения следующих задач [ ].

Во-первых, требуется установить вид операторов преобразования F, используемых в графодинамических уравнениях (9.8) и (9.10). Эта задача может быть сформулирована, в частности, как задача аппроксимации графовой траектории, установленной экспериментально, с помощью целочисленных уравнений.

Во-вторых, необходима формализация “законов жизни” графовых структур, определяющих закономерности их перестройки. Эти законы определяются внутренними и внешними факторами. Внутренние факторы связаны со свойствами коллектива разработчиков как социальной организации, внешние – с директивным установлением, например, длительности проектирования системы.

В-третьих, требуется исследование поведения решения графодинамических уравнений.

### **8.2.9. Пример построения функционально-структурной**

### **модели архитектуры системы**

Рассмотрим систему, функциональная схема которой приведена на рис. 9.17. Система является реверсивной – функции “ввод” и “вывод” могут меняться местами и эти функции реализуются идентично, что является основанием рассматривать далее системную функцию П0 и составляющие её функции П1 и П2.

Естественным в данном случае будет решение использовать для реализации функции П1 автоматизированное рабочее место АРМ пользователя, а для функции П2 предусмотреть сервер. Комплекс средств автоматизации АРМ пользователя представим следующими видами обеспечений: технического - ТО1, программного – ПО1, а также, учитывая присутствие человека, кадровым – КО и эргономическим – ЭО. Комплекс средств автоматизации, реализующий функцию П2, будет включать обеспечения: техническое – ТО2, программное – ПО2, а также, учитывая, что здесь осуществляется обработка информации, информационное – ИО и алгоритмическое – АО, определяющие характер преобразований информации, задаваемый пользователем.

Декомпозиция системы, упорядочивающая функции и реализующие их обеспечения, приведена на рис. 9.18 и соответствующий этой декомпозиции функционально-структурный граф системы – на рис. 8.19.

Далее введём метрику в предположении, что функциональный или структурный (конструктивный) элемент, соответствующий конкретной вершине графа или декомпозиционной схемы, обладает несколькими свойствами, каждое из которых может быть оценено одним скалярным показателем. Это предположение далеко от истины, но необходимо для выяснения сути рассматриваемого вопроса.

Под **свойством** в данном случае понимается некоторый физический признак (индикатор), позволяющий отличать одно от другого различные возможные состояния объекта или объекты между собою.

Ниже приводятся принятые во внимание свойства, показатели для их оценки и единицы измерения показателей, соотнесённые конкретным вершинам декомпозиционной схемы (графа) на рис. 9.18 (рис.9.19).

Сначала рассмотрим систему в целом (функция П0). Будем исходить из правила **3π**, согласно которому система для пользователя должна быть понятной

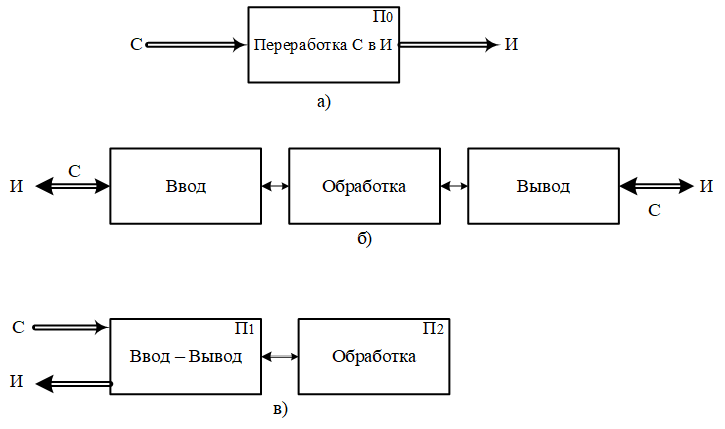


Рис.9.17. Функциональная схема системы

Обозначения: С – сведения,

И – информация,

П0 – основная производственная функция системы

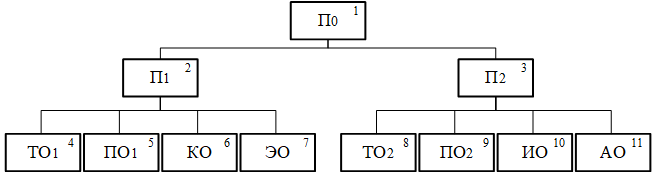


Рис.9.18. Декомпозиция системы

Обозначения:

ТО – техническое обеспечение,

ПО – программное обеспечение,

КО – кадровое обеспечение,

ЭО – эргономическое обеспечение

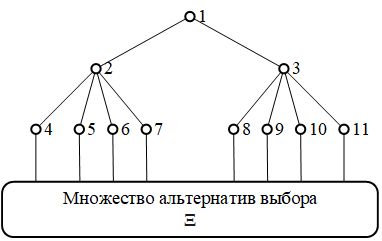


Рис.9.19. Функционально-структурный граф системы

(показатель для оценки этого свойства – **π1**), привлекательной (показатель – **π2**) и практичной (показатель – **π3**). Эти свойства определяют степень полезности системы для пользователя (показатель для оценки свойства полезности системы для пользователя –**πL**), т.е.

**πL = fπL(π1, π2,π3)**, (9.12)

где f**πL** – оператор свёртки.

Показатель**πL**имеет ясный физический смысл: чем полезнее система для пользователя, тем большую часть своей профессиональной деятельности он выполняет в среде системы. Этот показатель может быть оценён количественно.

Далее, функции П1 и П2 осуществляют ввод-вывод информации и её обработку соответственно. Оценим каждую из них двумя показателями: требуемым временем τ1 и τ2 (выражаются в единицах времени) и достоверностью выполнения δ1 и δ2 (выражаются вероятностью).

Техническое обеспечение ТО1 и ТО2 оценим быстродействием (тактовой частотой процессоров с поправкой на надёжность техники) –β1 и β2. Программное обеспечение ПО1 и ПО2 охарактеризуем количеством команд - q1и q2. Для кадрового обеспечения в лице пользователя определяющим является его профессиональная подготовка р, оцениваемая по лингвистической шкале: *рmin, …, pi, … ,pmax*. Рабочие условия на АРМ ℰ, оцениваемые по лингвистической шкале: ℰmin, …, ℰi, …, ℰmax, задаются эргономическим обеспечением ЭО. Информационное ИО и алгоритмическое АО обеспечения задаются соответственно физической моделью базы данных -μ и длиной алгоритма (числом шагов) -α.

Теперь, если введённые показатели соотнести вершинам функционально-структурного графа системы, то получим нелинейный граф показателей свойств системы, показанный на рис. 9.20. Чтобы воспользоваться этим графом, необходимо его линеаризовать, осуществив переход от частных показателей τ1, δ1 иτ2, δ2 к комплексным (интегральным) показателям △1 и △2:

△1 =f△1(τ1, δ1), (9.13)

△2 =f△2(τ2, δ2), (9.14)

где f△1, f△2 – операторы свёртки.

Физический смысл комплексных показателей в данном случае – это время реализации функций с гарантированной достоверностью.

Воспользовавшись комплексными показателями, получим линейный граф показателей свойств системы, представленный также на рис. 9.20. Этот граф задаёт множество проектных задач системы:

1. πL= *F𝜋L*(△1, △2), (15)

2. △1=*F*△1(β1,q1, *р,* ℰ), (16)

3. △2= *F*△2(α, μ, q2, β2), (17)

4. β1 = *F*𝛽1(Ξβ1), (18)

5. q1 =*F*q1(Ξq1), (19)

6. p = *F*p(Ξp), (20)

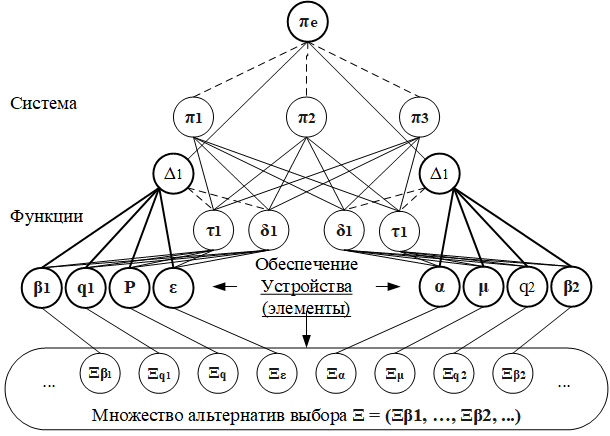


Рис.9.20. Граф показателей свойств системы

Обозначения:

 – нелинейный граф

 – линейный граф

 – вершины нелинейного графа

 – вершины линейного графа

 – связи, учитываемые операторами свертки

7. ℰ =*F*ε(Ξε), (21)

8. α= *F*α(Ξα), (22)

9. μ = *F*μ(Ξμ), (23)

10. q2 = *F*q2(Ξq2), (24)

11. β2  = *F*𝛽2(Ξβ2). (25)

Этому множеству соответствует граф проектных задач (далее по тексту – граф задач) – φ0(n), показанный на рис. 9.21.

Граф задач разрабатывается на организационном периоде жизненного цикла системы. На этом периоде принимается решение о создании системы, доказывается её реализуемость путем проведения необходимых научно-технических исследований и, возможно, опытно-конструкторских проработок, проводится технико-экономическое обоснование, включающее оценку потребных финансовых, материальных и интеллектуальных ресурсов и потенциала рынка информационных продуктов и услуг, и разрабатывается техническое задание. В итоге в распоряжении системных аналитиков оказывается достаточно информации для разработки исходного графа задач – φ0(n). Этот граф задаёт графовую траекторию системы на её жизненном цикле, показанную на рис. 9.22.

Производные графы φ1(n), φ2(n), … содержат те задачи, для решения которых на соответствующей стадии достаточно данных. В конце проектирования, на стадии испытаний система полностью определена, по всем задачам приняты проектные решения и граф φ5(n) отображается в φ0(n) и далее сам в себя.

Граф комплексных показателем свойств системы и соответствующее ему множество проектных задач является графовой моделью АИС как объекта проектирования. Графовая модель в процессе проектирования системы при переходе от текущей стадии к следующей видоизменяется за счет того, что накапливаются знания о системе. Поэтому решенные проектные задачи исключаются из модели, а задачи, для которых появились исходные данные, актуализируются. Таким образом, жизненный цикл системы являет собой графовую траекторию, которая является графодинамической моделью процесса проектирования как объекта автоматизации.

Графовую и графодинамическую модели разрабатывают системные аналитики, которые тем самым формируют изначально архитектуру создаваемой системы и формулируют исходные данные для узких специалистов: инженеров, программистов, инженерных психологов и др., каждый из которых решает задачи в пределах своей компетенции с максимальной пользой для будущей автоматизированной информационной системы.

Формирование метрики системы и определение операторов **fi** проектных задач является серьёзной научно-технической проблемой.

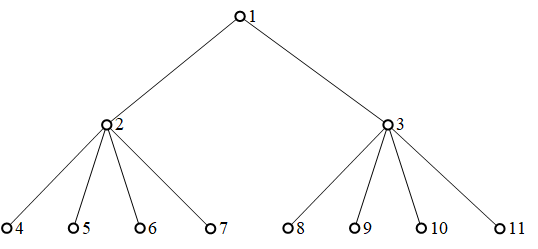


Рис.9.21. Граф проектных задач – φ0(n)

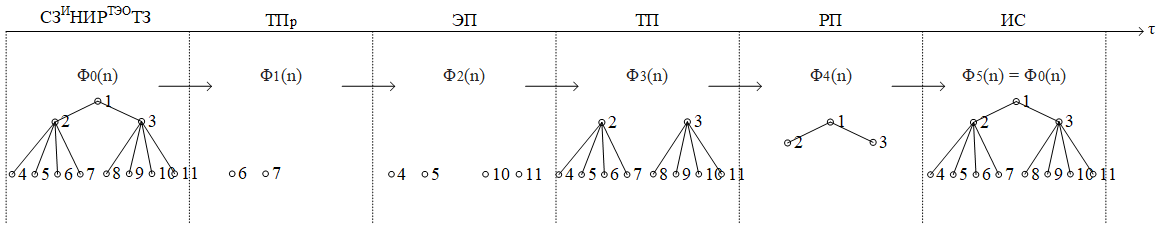


Рис. 9.22. Функционально-структурная модельсистемы

Обозначения: ТПр – техническое предложение

ЭП – эскизный проект

ТП – технический проект

РП – рабочий проект

ИС – испытания

СЗ – социальный заказ

НИР – научно-исследовательская работа

ТЭО – технико-экономическое обоснование

ТЗ – техническое задание

τ– время

**Контрольные вопросы для самопроверки**

**лекционного материала**