Лекція

Моделювання бізнес-процесів

Моделювання ділових процесів, як правило, виконується за допомогою case-засобів. До таких засобів відносяться BPwin (PLATINUM technology), Silverrun (Silverrun technology), Oracle Designer (Oracle), Rational Rose (Rational Software) і ін. Функціональні можливості інструментальних засобів структурного моделювання ділових процесів будуть розглянуті на прикладі case-засоби BPwin.

BPwin підтримує три методології моделювання: функціональне моделювання (IDEF0); опис бізнес-процесів (IDEF3); діаграми потоків даних (DFD).

Інструментальне середовище BPwin

BPwin має досить простий і інтуїтивно зрозумілий інтерфейс користувача. При запуску BPwin за замовчуванням з'являється основна панель інструментів, палітра інструментів (вид якої залежить від обраної нотації) і, в лівій частині, навігатор моделі - Model Explorer.

При створенні нової моделі виникає діалог, в якому слід вказати, чи буде створена модель заново або вона буде відкрита з файлу або з репозиторію ModelMart, потім внести ім'я моделі і вибрати методологію, в якій буде побудована модель.

Як було зазначено вище, BPwin підтримує три методології - IDEF0, IDEF3 і DFD, кожна з яких вирішує свої специфічні завдання. У BPwin можлива побудова змішаних моделей, т. Е. Модель може містити одночасно діаграми як IDEF0, так і IDEF3 і DFD. Склад палітри інструментів змінюється автоматично, коли відбувається перемикання з однієї нотації на іншу.

Модель в BPwin розглядається як сукупність робіт, кожна з яких оперує з деяким набором даних. Робота зображується у вигляді прямокутників, дані - у вигляді стрілок. Якщо клацнути по будь-якому об'єкту моделі лівою кнопкою миші, з'являється контекстне меню, кожен пункт якого відповідає редактору якої-небудь властивості об'єкта.

Побудова моделі IDEF0

На початкових етапах створення ІС необхідно зрозуміти, як працює організація, яку збираються автоматизувати. Керівник добре знає роботу в цілому, але не в змозі вникнути в деталі роботи кожного рядового співробітника. Рядовий співробітник добре знає, що діється на його робочому місці, але може не знати, як працюють колеги. Тому для опису роботи підприємства необхідно побудувати модель, яка буде адекватна предметної області та містити в собі знання всіх учасників бізнес-процесів організації.

Найбільш зручним мовою моделювання бізнес-процесів є IDEF0, де система представляється як сукупність взаємодіючих робіт або функцій. Така чисто функціональна орієнтація є принциповою - функції системи аналізуються незалежно від об'єктів, якими вони оперують. Це дозволяє більш чітко змоделювати логіку і взаємодію процесів організації.

Процес моделювання системи в IDEF0 починається зі створення контекстної діаграми - діаграми найбільш абстрактного рівня опису системи в цілому, що містить визначення суб'єкта моделювання, цілі і точки зору на модель.

Під суб'єктом розуміється сама система, при цьому необхідно точно встановити, що входить в систему, а що лежить за її межами, іншими словами, визначити, що буде в подальшому розглядатися як компоненти системи, а що як зовнішній вплив. На визначення суб'єкта системи будуть істотно впливати позиція, з якої розглядається система, і мета моделювання - питання, на які побудована модель повинна дати відповідь. Іншими словами, на початку необхідно визначити область моделювання. Опис області як системи в цілому, так і її компонентів є основою побудови моделі. Хоча передбачається, що в ході моделювання область може коректуватися, вона повинна бути в основному сформульована спочатку, оскільки саме область визначає напрям моделювання. При формулюванні області необхідно враховувати два компоненти - широту і глибину. Широта має на меті визначення меж моделі - що буде розглядатися всередині системи, а що зовні. Глибина визначає, на якому рівні деталізації модель є завершеною. При визначенні глибини системи необхідно пам'ятати про обмеження часу - трудомісткість побудови моделі росте в геометричній прогресії зі збільшенням глибини декомпозиції. Після визначення меж моделі передбачається, що нові об'єкти не повинні вноситися в моделируемую систему.

Мета моделювання визначається з відповідей на наступні питання:

Чому цей процес повинен бути змодельований?

Що повинна показувати модель?

Що може отримати клієнт?

Точка зору (Viewpoint).

Під точкою зору розуміється перспектива, з якою спостерігалася система при побудові моделі. Хоча при побудові моделі враховуються думки різних людей, всі вони повинні дотримуватися єдиної точки зору на модель. Точка зору повинна відповідати меті і кордонів моделювання. Як правило, вибирається точка зору людини, відповідального за моделируемую роботу в цілому.

IDEF0-модель передбачає наявність чітко сформульованої мети, єдиного суб'єкта моделювання і однієї точки зору. Для внесення області, цілі і точки зору в моделі IDEF0 в BPwin слід вибрати пункт меню Model / Model Properties, що викликає діалог Model Properties. У закладці Purpose слід внести мета і точку зору, а в закладку Definition - визначення моделі і опис області.

У закладці Status того ж діалогу можна описати статус моделі (чорновий варіант, робочий, остаточний і т. Д.), Час створення і останнього редагування (відстежується надалі автоматично по системній датою). У закладці Source описуються джерела інформації для побудови моделі (наприклад, "Опитування експертів предметної області та аналіз документації"). Закладка General служить для внесення імені проекту і моделі, імені та ініціалів учасника і тимчасових рамок моделі - AS-IS і ТО-ВЕ.

Моделі AS-IS і ТО-ВЕ. Зазвичай спочатку будується модель існуючої організації роботи - AS-IS (як є). Аналіз функціональної моделі дозволяє зрозуміти, де знаходяться найбільш слабкі місця, в чому полягатимуть переваги нових бізнес-процесів і наскільки глибоким змінам піддасться існуюча структура організації бізнесу. Деталізація бізнес-процесів дозволяє виявити недоліки організації навіть там, де функціональність на перший погляд здається очевидною. Знайдені в моделі AS-IS недоліки можна виправити при створенні моделі ТО-ВЕ (як буде) - моделі нової організації бізнес-процесів.

Технологія проектування ІС має на увазі спочатку створення моделі AS-IS, її аналіз і поліпшення бізнес-процесів, тобто створення моделі ТО-ВЕ, і тільки на основі моделі ТО-ВЕ будується модель даних, прототип і потім остаточний варіант ІС.

Іноді поточна AS-IS і майбутня ТО-ВЕ моделі розрізняються дуже сильно, так що перехід від початкового до кінцевого стану стає неочевидним. У цьому випадку необхідна третя модель, що описує процес переходу від початкового до кінцевого стану системи, оскільки такий перехід - це теж бізнес-процес.

Результат опису моделі можна отримати в звіті Model Report. Діалог настройки звіту по моделі викликається з пункту меню Tools / Reports / Model Report.

У діалозі настройки слід вибрати необхідні поля, при цьому автоматично відображається черговість виведення інформації в звіт.

Основу методології IDEF0 складає графічна мова опису бізнес-процесів. Модель в нотації IDEF0 являє собою сукупність ієрархічно впорядкованих і взаємопов'язаних діаграм. Кожна діаграма є одиницею опису системи і розташовується на окремому аркуші.

Модель може містити чотири типи діаграм:

контекстну діаграму (у кожній моделі може бути тільки одна контекстна діаграма);

діаграми декомпозиції;

діаграми дерева вузлів;

діаграми тільки для експозиції (FEO).

Контекстна діаграма є вершиною деревовидної структури діаграм і являє собою саме загальний опис системи та її взаємодії із зовнішнім середовищем. Після опису системи в цілому проводиться розбиття її на великі фрагменти. Цей процес називається функціональної декомпозицією, а діаграми, які описують кожен фрагмент і взаємодія фрагментів, називаються діаграмами декомпозиції. Після декомпозиції контекстної діаграми проводиться декомпозиція кожного великого фрагмента системи на більш дрібні і так далі, до досягнення потрібного рівня подробиці опису. Після кожного сеансу декомпозиції проводяться сеанси експертизи - експерти предметної області вказують на відповідність реальних бізнес-процесів створеним діаграм. Знайдені невідповідності виправляються, і тільки після проходження експертизи без зауважень можна приступати до наступного сеансу декомпозиції. Так досягається відповідність моделі реальним бізнес-процесів на будь-якому і кожному рівні моделі. Синтаксис опису системи в цілому і кожного її фрагмента однаковий у всій моделі.

Діаграма дерева вузлів показує ієрархічну залежність робіт, але не взаємозв'язку між роботами. Діаграм дерев вузлів може бути в моделі як завгодно багато, оскільки дерево може бути побудовано на довільну глибину і не обов'язково з кореня.

діаграми для експозиції (FEO) будуються для ілюстрації окремих фрагментів моделі, для ілюстрації альтернативної точки зору, або для спеціальних цілей.

Роботи (Activity) позначають зазначені процеси, функції або завдання, які відбуваються протягом певного часу і мають розпізнавані результати. Роботи зображуються у вигляді прямокутників. Всі роботи повинні бути названі і визначені. Ім'я роботи повинне бути виражене віддієслівним іменником, що позначає дію (наприклад, "Діяльність компанії", "Прийом замовлення" і т.д.). Робота "Діяльність компанії" може мати, наприклад, наступне визначення: "Це навчальна модель, що описує діяльність компанії". При створенні нової моделі (меню File / New) автоматично створюється контекстна діаграма з єдиною роботою, яка зображує систему в цілому.

Діаграми декомпозиції містять споріднені роботи, т. Е. Дочірні роботи, що мають загальну батьківську роботу. Для створення діаграми декомпозиції слід клацнути по кнопці на панелі інструментів.

Виникає діалог Activity Box Count, в якому слід вказати нотацію нової діаграми і кількість робіт на ній. Зупинимося поки на нотації IDEF0 і клацнемо на ОК. З'являється діаграма декомпозиції. Допустимий інтервал числа робіт - 2-8. Декомпозіровать роботу на одну роботу не має сенсу: діаграми з кількістю робіт більше восьми виходять перенасиченим і погано читаються. Для забезпечення наочності і кращого розуміння модельованих процесів рекомендується використовувати від трьох до шести блоків на одній діаграмі.

Якщо виявляється, що кількість робіт недостатньо, то роботу можна додати в діаграму, клацнувши спочатку по кнопці на палітрі інструментів, а потім по вільному місцю на діаграмі.

Роботи на діаграмах декомпозиції зазвичай розташовуються по діагоналі від лівого верхнього кута до правого нижнього.

Такий порядок називається порядком домінування. Згідно з цим принципом розташування в лівому верхньому кутку поміщається найважливіша робота або робота, виконувана за часом першої. Далі вправо вниз розташовуються менш важливі або виконуються пізніше роботи. Таке розміщення полегшує читання діаграм, крім того, на ньому ґрунтується поняття взаємозв'язків робіт (див. Нижче).

Кожна з робіт на діаграмі декомпозиції може бути в свою чергу декомпозирована. На діаграмі декомпозиції роботи нумеруються автоматично зліва направо. Номер роботи показується в правому нижньому кутку. У лівому верхньому кутку зображається невелика діагональна риса, яка показує, що дана робота не була декомпозирована.

Стрілки (Arrow) описують взаємодію робіт і являють собою якусь інформацію, виражену іменниками. (Наприклад, "Дзвінки клієнтів", "Правила і процедури", "Бухгалтерська система".)

В IDEF0 розрізняють п'ять типів стрілок:

Вхід (Input) - матеріал або інформація, які використовуються або перетворюються роботою для отримання результату (виходу). Допускається, що робота може не мати жодної стрілки входу. Кожен тип стрілок підходить до певної стороні прямокутника, що зображує роботу, або виходить з неї. Стрілка входу малюється як що входить в ліву грань роботи. При описі технологічних процесів (для цього і був придуманий IDEF0) не виникає проблем визначення входів. При моделюванні ІС, коли стрілками є не фізичні об'єкти, а дані, не все так очевидно. Наприклад, при "Прийомі пацієнта" карта пацієнта може бути і на вході і на виході, тим часом якість цих даних змінюється. Іншими словами, в нашому прикладі для того, щоб виправдати своє призначення, стрілки входу і виходу повинні бути точно визначені з тим, щоб вказати на те, що дані дійсно були перероблені (наприклад, на виході - "Заповнена карта пацієнта"). Дуже часто складно визначити, чи є дані входом або управлінням. В цьому випадку підказкою може служити інформація про те, переробляються / чи змінюються дані в роботі чи ні. Якщо змінюються, то, швидше за все, це вхід, якщо немає - управління.

Управління (Control) - правила, стратегії, процедури або стандарти, якими керується робота. Кожна робота повинна мати хоча б одну стрілку управління. Стрілка управління малюється як що входить в верхню межу роботи. Управління впливає на роботу, але не перетвориться роботою. Якщо мета роботи - змінити процедуру або стратегію, то така процедура або стратегія буде для роботи входом. У разі виникнення невизначеності в статусі стрілки (управління або увійти) рекомендується малювати стрілку управління.

Вихід (Output) - матеріал або інформація, які виробляються роботою. Кожна робота повинна мати хоча б одну стрілку виходу. Робота без результату не має сенсу і не повинна моделюватися. Стрілка виходу малюється як виходить з правої межі роботи.

Механізм (Mechanism) - ресурси, які виконують роботу, наприклад персонал підприємства, верстати, пристрої і т. Д. Стрілка механізму малюється як що входить в нижню межу роботи. На розсуд аналітика стрілки механізму можуть не зображатися в моделі.

Виклик (Call) - спеціальна стрілка, що вказує на іншу модель роботи. Стрілка виклику малюється як виходить із нижньої межі роботи. Стрілка виклику використовується для вказівки того, що деяка робота виконується за межами модельованої системи. У BPwin стрілки виклику використовуються в механізмі злиття і розділення моделей.

Граничні стрілки. Стрілки на контекстній діаграмі служать для опису взаємодії системи з навколишнім світом. Вони можуть починатися біля кордону діаграми і закінчуватися у роботи, або навпаки. Такі стрілки називаються граничними.

Для внесення граничної стрілки входу слід:

клацнути по кнопці з символом стрілки;

в палітрі інструментів перенести курсор до лівої сторони екрану, поки не з'явиться початкова штрихова смужка;

клацнути один раз по смужці (звідки виходить стрілка) і ще раз в лівій частині роботи з боку входу (де закінчується стрілка);

повернутися в палітру інструментів і вибрати опцію редагування стрілки;

натиснути правою кнопкою миші на лінії стрілки, у спливаючому меню вибрати Name і додати ім'я стрілки в закладці Name діалогу IDEF0 Arrow Properties.

Стрілки управління, входу, механізму і виходу зображаються аналогічно. Імена знову внесених стрілок автоматично заносяться в словник Arrow Dictionary.

ICOM-коди. Діаграма декомпозиції призначена для деталізації роботи. На відміну від моделей, що відображають структуру організації, робота на діаграмі верхнього рівня в IDEF0 - це не елемент управління нижчестоящими роботами. Роботи нижнього рівня - це те ж саме, що роботи верхнього рівня, але в більш детальному викладі. Як наслідок цього кордону роботи верхнього рівня - це те ж саме, що кордони діаграми декомпозиції. ICOM (абревіатура від Input, Control, Output і Mechanism) - коди, призначені для ідентифікації граничних стрілок. Код ICOM містить префікс, який відповідає типу стрілки (I, С, О або М), і порядковий номер.

BPwin вносить ICOM-коди автоматично. Для відображення ICOM-кодів слід включити опцію ICOM codes на закладці Display діалогу Model Properties (меню Model / Model Properties).

Словник стрілок редагується за допомогою спеціального редактора Arrow Dictionary Editor, в якому визначається стрілка і вноситься відноситься до неї коментар. Словник стрілок вирішує дуже важливу задачу. Діаграми створюються аналітиком для того, щоб провести сеанс експертизи, т. Е. Обговорити діаграму з фахівцем предметної області. У будь-якій предметній області формується професійний жаргон, причому дуже часто жаргонні вирази мають нечіткий сенс і сприймаються різними фахівцями по-різному. У той же час аналітик - автор діаграм повинен вживати ті вирази, які найбільш зрозумілі експертам. Оскільки формальні визначення часто складні для сприйняття, аналітик змушений вживати професійний жаргон, а щоб не виникло неоднозначних трактувань, в словнику стрілок кожному поняттю можна дати розширене і, якщо це необхідно, формальне визначення.

Вміст словника стрілок можна роздрукувати у вигляді звіту (меню Tools / Report / Arrow Report ...) і отримати тлумачний словник термінів предметної області, що використовуються в моделі.

Незв'язані граничні стрілки (unconnected border arrow). При декомпозиції роботи входять в неї і виходили з неї стрілки (крім стрілки виклику) автоматично з'являються на діаграмі декомпозиції (міграція стрілок), але при цьому не стосуються робіт. Такі стрілки називаються незв'язаними і сприймаються в BPwin як синтаксична помилка.

Для зв'язування стрілок входу, управління або механізму необхідно перейти в режим редагування стрілок, клацнути по наконечнику стрілки і потім по відповідному сегменту роботи. Для зв'язування стрілки виходу необхідно перейти в режим редагування стрілок, клацнути по сегменту виходу роботи і потім по стрілці.

Внутрішні стрілки. Для зв'язку робіт між собою використовуються внутрішні стрілки, тобто стрілки, які не стосуються кордону діаграми, починаються у однієї і закінчуються в іншої роботи.

Для малювання внутрішньої стрілки необхідно в режимі малювання стрілок клацнути по сегменту (наприклад, виходу) однієї роботи і потім по сегменту (наприклад, входу) інший. В IDEF0 розрізняють п'ять типів зв'язків робіт.

Зв'язок по входу (output-input), коли стрілка виходу вищестоящої роботи (далі - просто вихід) прямує на вхід нижчої.

Зв'язок з управління (output-control), коли вихід вищестоящої роботи прямує на управління нижчої. Зв'язок з управління показує домінування вищестоящої роботи. Дані або об'єкти виходу вищестоящої роботи не змінюються в нижчої.

Зворотній зв'язок по входу (output-input feedback), коли вихід нижчої роботи направляється на вхід вищестоящої. Такий зв'язок, як правило, використовується для опису циклів.

Зворотній зв'язок з управління (output-control feedback), коли вихід нижчої роботи спрямовується на управління вищестоящої. Зворотній зв'язок з управління часто свідчить про ефективність бізнес-процесу.

Зв'язок вихід-механізм (output-mechanism), коли вихід однієї роботи спрямовується на механізм інший.

Явні стрілки. Явна стрілка має джерелом одну-єдину роботу і призначенням теж одну-єдину роботу.

Розгалужуються і зливаються стрілки. Одні і ті ж дані або об'єкти, породжені однією роботою, можуть використовуватися відразу в декількох інших роботах. З іншого боку, стрілки, породжені в різних роботах, можуть являти собою однакові або однорідні дані або об'єкти, які в подальшому використовуються або переробляються в одному місці. Для моделювання таких ситуацій в IDEF0 використовуються розгалужуються і зливаються стрілки. Для розгалуження стрілки потрібно в режимі редагування стрілки клацнути по фрагменту стрілки і за відповідним сегменту роботи. Для злиття двох стрілок виходу потрібно в режимі редагування стрілки спочатку клацнути по сегменту виходу роботи, а потім по відповідному фрагменту стрілки.

Сенс розгалужуються і зливаються стрілок передається ім'ям кожної гілки стрілок. Існують певні правила іменування таких стрілок. Розглянемо їх на прикладі розгалужуються стрілок. Якщо стрілка іменована до розгалуження, а після розгалуження жодна з гілок не йменувався, то мається на увазі, що кожна гілка моделює ті ж дані або об'єкти, що і гілка до розгалуження.

Якщо стрілка іменована до розгалуження, а після розгалуження котрась із гілок теж іменована, то мається на увазі, що ці гілки відповідають іменування. Якщо при цьому будь-яка гілка після розгалуження залишилася неіменованого, то мається на увазі, що вона моделює ті ж дані або об'єкти, що і гілка до розгалуження.

Неприпустима ситуація, коли стрілка до розгалуження не йменувався, а після розгалуження не йменувався котрась із гілок. BPwin визначає таку стрілку як синтаксичну помилку.

Правила іменування зливаються стрілок повністю аналогічні - помилкою буде вважатися стрілка, яка після злиття немає іменована, а до злиття немає іменована яка-небудь з її гілок. Для іменування окремої гілки розгалужуються і зливаються стрілок слід виділити на діаграмі тільки одну гілку, після чого викликати редактор імені та присвоїти ім'я стрілкою. Це ім'я буде відповідати тільки виділеної гілки.

Туннелирование стрілок. Знову внесені граничні стрілки на діаграмі декомпозиції нижнього рівня зображуються в квадратних дужках і автоматично не з'являються на діаграмі верхнього рівня.

Туннелирование може бути застосовано для зображення малозначущих стрілок. Якщо на будь-якої діаграмі нижнього рівня необхідно зобразити малозначущі дані або об'єкти, які не обробляються або не використовуються роботами на поточному рівні, то їх необхідно направити на вищестоящий рівень (на батьківську діаграму). Якщо ці дані не використовуються на батьківській діаграмі, їх потрібно направити ще вище, і т. Д. В результаті малозначима стрілка буде зображена на всіх рівнях і утруднить читання всіх діаграм, на яких вона присутня. Виходом є туннелирование стрілки на найнижчому рівні. Таке туннелирование називається «не-в-батьківської-діаграмі".

Іншим прикладом тунелювання може бути ситуація, коли стрілка механізму мігрує з верхнього рівня на нижній, причому на нижньому рівні цей механізм використовується однаково у всіх роботах без винятку. (Передбачається, що не потрібно деталізувати стрілку механізму, т. Е. Стрілка механізму на дочірній роботі іменована до розгалуження, а після розгалуження гілки не мають власного імені). В цьому випадку стрілка механізму на нижньому рівні може бути видалена, після чого на батьківській діаграмі вона може бути тунелюватись, а в коментарі до стрілки або в словнику можна вказати, що механізм буде використовуватися у всіх роботах дочірньої діаграми декомпозиції. Таке туннелирование називається «не-в-дочірньої-роботі".

Нумерація робіт і діаграм. Всі роботи моделі нумеруються. Номер складається з префікса і числа. Може бути використаний префікс будь-якої довжини, але зазвичай використовують префікс А. Контекстна (коренева) робота дерева має номер А0. Роботи i декомпозиції А0 мають номера А1, А2, A3 і т. Д. Роботи декомпозиції нижнього рівня мають номер батьківської роботи і черговий порядковий номер, наприклад роботи декомпозиції A3 матимуть номери А31, А32, АЗЗ, А34 і т. Д. Роботи утворюють ієрархію, де кожна робота може мати одну батьківську і кілька дочірніх робіт, утворюючи дерево. Таке дерево називають деревом вузлів, а вищеописану нумерацію - нумерацією по вузлах. Діаграми IDEF0 мають подвійну нумерацію. По-перше, діаграми мають номера по вузлу. Контекстна діаграма завжди має номер А0, декомпозиція контекстної діаграми - номер А0, решта діаграми декомпозиції - номери за відповідним вузлу (наприклад, A1, A2, А21, А213 і т. Д.). BPwin автоматично підтримує нумерацію по вузлах, т. Е. При проведенні декомпозиції створюється нова діаграма і їй автоматично присвоюється відповідний номер. В результаті проведення експертизи діаграми можуть уточнюватися і змінюватися, отже, можуть бути створені різні версії однієї і тієї ж (з точки зору її розташування в дереві вузлів) діаграми декомпозиції. BPwin дозволяє мати в моделі тільки одну діаграму декомпозиції в даному вузлі. Колишні версії діаграми можна зберігати у вигляді паперової копії або як FEO-діаграму. (На жаль, при створенні FEO-діаграм відсутня можливість відкоту, т. Е. З діаграми можна отримати декомпозиції FEO, але не навпаки.) У будь-якому випадку слід відрізняти різні версії однієї і тієї ж діаграми. Для цього існує спеціальний номер - C-number, який повинен присвоюватися автором моделі вручну. C-number - це довільна рядок, але рекомендується дотримуватися стандарту, коли номер складається з літерного префікса і порядкового номера, причому в якості префікса використовуються ініціали автора діаграми, а порядковий номер відстежується автором вручну, наприклад МСВ00021.

Можливість злиття і розщеплення моделей забезпечує колективну роботу над проектом. Так, керівник проекту може створити декомпозицію верхнього рівня і дати завдання аналітикам продовжити декомпозицію кожній гілці дерева у вигляді окремих моделей. Після закінчення роботи над окремими гілками все подмодели можуть бути злиті в єдину модель. З іншого боку, окрема гілка моделі може бути відщеплена для використання в якості незалежної моделі, для доопрацювання або архівування.

BPwin використовує для злиття і розгалуження моделей стрілки виклику. Для злиття необхідно виконати наступні умови:

Обидві зливаються моделі повинні бути відкриті в BPwin.

Ім'я моделі-джерела, яке приєднують до моделі-цілі, має збігатися з ім'ям стрілки виклику роботи в моделі-цілі.

Стрілка виклику повинна виходити з недекомпозіруемой роботи (робота повинна мати діагональну межу в лівому верхньому кутку).

Імена контекстної роботи під'єднується моделі-джерела і роботи на моделі-цілі, до якої ми приєднуємо модель-джерело, повинні збігатися.

Модель-джерело повинна мати, принаймні, одну діаграму декомпозиції.

Для злиття моделей потрібно натиснути правою кнопкою миші по роботі зі стрілкою виклику в моделі-цілі і у спливаючому меню вибрати пункт Merge Model.

З'являється діалог, в якому слід вказати опції злиття моделі. При злитті моделей об'єднуються і словники стрілок і робіт. У разі однакових визначень можлива перезапис визначень або прийняття ухвал з моделі-джерела. Те саме можна сказати до імен стрілок, сховищ даних і зовнішніх посиланнях. (Сховища даних і зовнішні посилання - об'єкти діаграм потоків даних, DFD, будуть розглянуті нижче.)

Після підтвердження злиття (кнопка OK) модель-джерело під'єднується до моделі-цілі, стрілка виклику зникає, а робота, від якої відходила стрілка виклику, стає декомпозіруемой - до неї приєднується діаграма декомпозиції першого рівня моделі-джерела. Стрілки, що стосуються роботи на діаграмі моделі-цілі, автоматично НЕ мігрують в декомпозицію, а відображаються як недозволені. Їх слід туннелировать вручну.

У процесі злиття модель-джерело залишається незмінною, і до моделі-цілі підключається фактично її копія. Не потрібно плутати злиття моделей з синхронізацією. Якщо в подальшому модель-джерело буде редагуватися, ці зміни автоматично не потраплять в відповідну гілку моделі-цілі.

Поділ моделей проводиться аналогічно. Для відщеплення гілки від моделі слід натиснути правою кнопкою миші по декомпозіровать роботі (робота не повинна мати діагональної риси в лівому верхньому кутку) і вибрати у спливаючому меню пункт Split Model. У діалозі Split Options слід вказати ім'я створюваної моделі. Після підтвердження розщеплення в старій моделі робота стане недекомпозірованной (ознака - діагональна риса в лівому верхньому куті), буде створена стрілка виклику, її ім'я буде збігатися з ім'ям нової моделі, і, нарешті, буде створена нова модель, причому ім'я контекстної роботи буде збігатися з ім'ям роботи, від якої була "відірвана" декомпозиція.

Створення звітів в BPwin

BPwin має потужний інструмент генерації звітів. Звіти по моделі викликаються з пункту меню Report. Всього є сім типів звітів:

Model Report. Включає інформацію про контекст моделі - ім'я моделі, точку зору, область, мета, ім'я автора, дату створення та ін.

Diagram Report. Звіт по конкретної діаграмі. Включає список об'єктів (робіт, стрілок, сховищ даних, зовнішніх посилань і т. Д.).

Diagram Object Report. Найбільш повний звіт по моделі. Може включати повний список об'єктів моделі (робіт, стрілок із зазначенням їх типу та ін.) І властивості, що визначаються користувачем.

Activity Cost Report. Звіт про результати вартісного аналізу. Буде розглянуто нижче.

Arrow Report. Звіт по стрілках. Може містити інформацію зі словника стрілок, інформацію про роботу-джерелі, роботі-призначення стрілки і інформацію про розгалуженні і злиття стрілок.

Data Usage Report. Звіт про результати зв'язування моделі процесів і моделі даних. (Буде розглянуто нижче.)

Model Consistency Report. Звіт, що містить список синтаксичних помилок моделі.