Московский государственный университет им. М. В. Ломоносова, факультет психологии, кафедра психологии труда и инженерной психологии

Теория деятельности и разработка информационных систем: современное состояние, тенденции и перспективы

Курсовая работа выполнена студентом
202 группы II курса дневного отделения
Дегтяренко Иваном Александровичем.

Научный руководитель: Бурмистров И. В.
Рецензент: Орёл Е. А.

Москва, 2005.

Введение

С конца восьмидесятых годов прошлого века в разработке информационных систем появились достаточно серьёзные попытки применения теории деятельности. Изначально эта область исследований начала развиваться в странах Скандинавии, а пионерами в этой области были такие учёные, как Иржо Энгестрём (Yrjö Engeström), Кари Кутти (Kari Kuutti) и Сюзанна Бодкер (Susanne Bødker). Теория деятельности стала одним из перспективных новых подходов как в практике разработки информационных систем, так и в теоретической дисциплине, называемой в англоязычной литературе «взаимодействием человека и компьютера» (Human-Computer Interaction, далее HCI).

Здесь термин «информационная система» подразумевает некоторую совокупность людей, технических устройств и методов работы с информацией: её сбора, обработки, передачи, распространения и т. д. [25] В современной действительности такие системы очень часто компьютеризированы. Однако можно привести пример и некомпьютеризированной системы, такой как библиотека: и сотрудники библиотеки, и алфавитный каталог, и правила поиска книг и пользования ими — это части библиотеки как информационной системы.

В изучении HCI традиционно используется исследовательская парадигма когнитивной психологии. Однако к концу восьмидесятых годов многими исследователями был замечен кризис данной области [4, 34, 46]. Когнитивный подход оказался подходящим далеко не для всех целей, и в HCI стали появляться новые подходы, такие как этнографический подход [38], теория деятельности, теория сетей субъектов (actor-network theory) [35].

В отличие от когнитивного подхода, разработки в рамках которого в основном используются только при построении интерфейсов «человек-машина», новые подходы стремятся обеспечить построение всей информационной системы. Они стремятся сделать разработки не просто удобными в ис-

пользовании, но и максимально полезными [39], так как эти теории позволяют гораздо более глубоко анализировать потребности пользователей и контекст их работы, изучая реальную деятельность в её естественной среде [38].

Многие исследователи представляют теорию деятельности как полезный описательный инструмент [10, 39], позволяющий чётко определить структуру деятельности, а также суть имеющихся в системе проблем. Такой инструмент незаменим в изучении совместной деятельности и создании многопользовательских программ. Область HCI, посвящённая этой проблематике, называется «компьютерной поддержкой совместного труда» (Computer-Supported Communicative Work, далее CSCW) [23]. Эта область стала особенно активно развиваться одновременно с появлением новых подходов в HCI.

Теория деятельности не опровергает достижения когнитивной психологии и не противостоит ей. Одна из основных представителей данного направления американка Бонни Нарди пишет о новом, «пост-постмодернистском голосе в HCI», который стремится не к спорам, а к непротиворечивому использованию результатов как номотетического (когнитивного), так и идеографических подходов, которые могут быть объединены в единую науку о взаимоотношении человека и компьютера при помощи понятийного аппарата теории деятельности [39].

На данный момент можно констатировать наличие международного междисциплинарного сообщества исследователей в рамках деятельностного подхода [34], которые, начиная с 1986 года, раз в 4 года проводят международные конференции и издают собственный журнал. В основном, за пределами России теория деятельности разрабатывается в Финляндии, Норвегии, Великобритании, США и Австралии. Некоторые статьи также написаны авторами из таких стран, как Германия и Италия. В 2004 году в Копенгагене был проведён первый международный семинар ATIT по практическим методам разработки информационных систем, основанным на теории деятельности. Также по данной тематике вышли сборник статей под редакцией Б. Нарди [15] и 3 выпуска сборника, выпущенного австралийским

университетом из города Воллонгонг [26, 27, 28], а также несколько специальных выпусков тематических журналов [14, 45]. Также на базе университета Воллонгонг существует Activity Theory Usability Lab — компания по разработке информационных систем, которая позиционируют себя как лабораторию, основывающую свою работу на теории деятельности. Среди российских исследователей данной проблемой занимаются только В. Н. Каптелинин, Г. З. Бедный и И. Вереникина, а также, в некоторой степени, Б. М. Величковский, которые на данный момент работают за рубежом. Однако на родине теории деятельности должны открываться большие перспективы для развития данной области, так как она хорошо подходит для прикладного использования деятельностного подхода, а разработка программного обеспечения в России достаточно быстро развивается.

Обзор литературы

История проблемы

Когда компьютеры только появились, их пользователями были учёные. Часто программистом и единственным пользователем программы был один и тот же человек. В то время не возникала проблема удобства использования. Затем компьютеры появились в крупных коммерческих компаниях, и постепенно стала развиваться коммерческая разработка программного и аппаратного обеспечения. Так как пользователи становились всё менее осведомлёнными, а программы — многопользовательскими и большими по объёму кода, стали появляться проблемы с пониманием программ пользователями и с необходимостью координации целой команды разработчиков программы. В этот момент в компьютерных науках и стали развиваться такие направления, как HCI и ПО-инженерия (Software engineering) [6]. Первое направлено на создание удобных для пользователей, полезных и эффективных программ, а второе — на усовершенствование процесса их разработки.

С развитием классического HCI в рамках когнитивного подхода появилось множество фундаментальных руководств, которые обычно содержали одну-две главы, описывающих теоретические основы когнитивной психологии. При этом, однако, эти главы почти не увязывались с остальным содержанием книг [34]. Эта связь ограничивалась советами вроде: «Учитывайте ограниченные возможности человеческой системы переработки информации». В основном же дисциплина вырождалась в издание многочисленных сборников эмпирических «руководящих принципов» (guidelines) на все случаи жизни (например, [42]). В области ПО-инженерии теория также в основном ограничивалась достаточно банальными вещами, вроде схем цикла разработки программного обеспечения. В основном, теория могла не порождать эти принципы, а лишь до некоторой степени комментировать их. Так, например, идея «прямого манипулирования» (наиболее известным примером применения которой на практике является приём «потяни и брось» (drag'n'drop), используемый в большинстве современных графических интерфейсов (Microsoft Windows™, Mac OS™, KDE™)) использовалась в течение примерно пятнадцати лет, пока не появилось хоть какого-то теоретического описания этого феномена [24]. Также было показано, что большинство разработчиков вовсе не знали о такой дисциплине, как HCI, и пользовались в первую очередь «здравым смыслом» и «руководящими принципами» [34].

В конце восьмидесятых годов в сообществе исследователей HCI проходили множественные дискуссии о смене парадигмы исследований. Многие критиковали даже самые основные достижения когнитивного подхода в HCI [4, 34, 46] и обсуждали направление необходимых изменений в области. Многие считали необходимым изучать взаимодействие человека и компьютера в более широком контексте. Экспериментальные исследования, при наличии точнейших методик и, соответственно, возможности оценивать количественные результаты и проверять достоверность гипотез, при возможности варьировать условия эксперимента и при всех других важнейших пре-

имуществах экспериментального подхода, имеют существенный недостаток. В огромном большинстве своём они сводят изучение информационных систем к изучению интерфейса «человек-компьютер», к изучению некоторой последовательности формальных действий человека с компьютером, к изучению обмена информацией между ними. В эксперименте теряется огромная доля информации, имеющейся в реальной трудовой практике.

Основной задачей новой методологии было «человечное» изучение информационных систем, когда первичным является не компьютер, а человек (интересно, что существуют различные названия для дисциплины: Human-Computer Interaction и Computer-Human Interaction). Эта идея как нельзя лучше выражается названием одной из статей Л. Бэннон (L. Bannon), которое в переводе с английского звучит примерно так: «От человеческого фактора к человеку» [5].

Попытки приблизить исследования HCI к реальности породили несколько новых подходов [38]. Так, убеждённые сторонники экспериментального подхода предлагали создавать так называемые «естественные» или «экологические» эксперименты. Другой новый подход представляют этнографические методы [22]. Этот подход позволяет более полно передать картину реальной трудовой практики, во многом, благодаря своей ориентации на социальные аспекты этой практики.

Как оказывается, ни один из этих методов не позволяет полностью раскрыть сущность какой-либо системы. Когнитивный подход не приспособлен к пониманию и описанию таких сущностей, как «контекст», «трудовая практика», «ситуация» [39], в когнитивной парадигме очень затруднено изучение социальных аспектов систем. Этнографический же подход не предоставляет точных методов и строгой терминологии. Словарь этнографических исследований достаточно расплывчат, что не позволяет чётко концептуализировать данные. По сути, терминология каждый раз разрабатывается заново для конкретной задачи («ad hoc») [40]. Поэтому, несмотря на то, что теория деятельности не даёт готовых практических инструментов для непосредственной

разработки систем, её понятийный аппарат представляется прекрасным инструментом описания и концептуализации. Позволяя в строгих терминах описать социальный и прочие аспекты реальной трудовой практики, она несёт в себе плюсы обоих описанных выше подходов и позволяет объединить факты, полученные ими, в единую структуру. Новый теоретический подход необходим для объединения множества разрозненных направлений исследования, теория деятельности даёт большие надежды на построение дисциплины HCI будущего.

Другая проблема когнитивного подхода состоит в его ориентированности на схемы, «ментальные модели», «когнитивные карты» и прочие виды психического отражения предметов, за которыми часто теряется собственно сущность вещей. Ещё в 1988 году Дональд Норман, являющийся большим авторитетом в области HCI, писал о необходимости исследования «вещей в их вещности» [44]. Теоретико-деятельностная концепция опосредствования решает также и проблему связи предмета психологии с изучением сущности вещей, рассматривая вещи как то, что опосредствует человеческую деятельность. Потерянная связь между психикой и внешней деятельностью человека наиболее адекватно отображается в рамках теории деятельности. Она позволяет избежать психофизического параллелизма. В зарубежной психологии уже была некоторая традиция соотнесения сознания и деятельности: на это в первую очередь указывал У. Найссер [41], бывший при этом одним из основателей когнитивной психологии. Это ещё раз указывает на то, что деятельностный подход не имеет своей целью противоречить предыдущим достижениям, в некоторой степени он возвращает когнитивных психологов к собственным корням.

Особенности трактовки теории деятельности в HCI-исследованиях

Теория деятельности трактуется зарубежными исследователями без принципиальных изменений или искажений. В первую очередь они опирают-

ся на переведённые тексты А. Н. Леонтьева, такие как «Деятельность. Сознание. Личность», «Проблемы развития психики» и статьи из журнала *Soviet Psychology* [36, 37], а также на статьи некоторых других российских классиков (например, [21]). Основной упор в работах исследователей HCI делается на такие понятия теории деятельности, как, например, элементы структуры деятельности, так как они имеют большие возможности практического использования в описании трудовой практики. Однако при этом они также говорят о необходимости не забывать о таких фундаментальных идеях, как, например, единство сознания и деятельности [39], а также упоминают о философских корнях теории деятельности, описывая некоторые идеи диалектического материализма [18, 32]. Характерной особенностью зарубежной трактовки теории деятельности является то, что в этих работах в первую очередь рассматривается коллективная деятельность.

В качестве основных понятий теории деятельности описываются такие, как «деятельность», «артефакт», «опосредствование», «действие», «операция», «мотив», «цель», «противоречие», «инструмент», «интериоризация» [10, 34]. Однако также исследователи вводят некоторые понятия, отсутствующие в классических трудах, такие как «уровни взаимодействия» [6], «сети деятельностей» [34], «разделение труда», «правила» [17, 34] и т. д. Терминология теории деятельности получила некоторое расширение в качестве инструмента описания структуры труда в рамках направления, называемого «развитийным исследованием труда» (developmental work research), которое в основном описывается в трудах И. Энгестрёма (Yrjö Engeström) и его коллег по работе в университете финского города Аархус [17, 19, 31]. Другой подобной надстройкой к теории деятельности — некоторым дополнением к ней, находящемся на уровне методологии конкретного направления науки — является системно-структурная теория деятельности, развиваемая российским исследователем Г. З. Бедным, работающим сейчас в колледже округа Эссекс, США [7, 8].

Основные понятия: деятельность

Деятельность представляет собой важнейшую единицу анализа, которая позволяет описать процессы, происходящие в любой информационной системе в виде сети деятельностей. Так, в примере, приведённом в статье К. Кутти [34] в компании, занимающейся разработкой программного обеспечения, выделяется деятельность разработчиков и менеджеров проектов, объектом которой является сама незавершённая система, которую необходимо превратить в полезное и надёжное приложение. Таким образом, цель разработчиков — «разработать систему». В другой выделяемой деятельности сообществом являются менеджеры проектов и их начальство, а объектом этой деятельности — финансовый статус компании. Цель этой деятельности — обеспечить максимально возможную прибыль компании. Третьей деятельностью является борьба менеджеров за более высокие карьерные посты, в которой «объектным полем» являются возможности и качества, которыми обладает каждый претендент в глазах комитета по отбору кадров. Здесь цель — получить более высокую должность в компании.

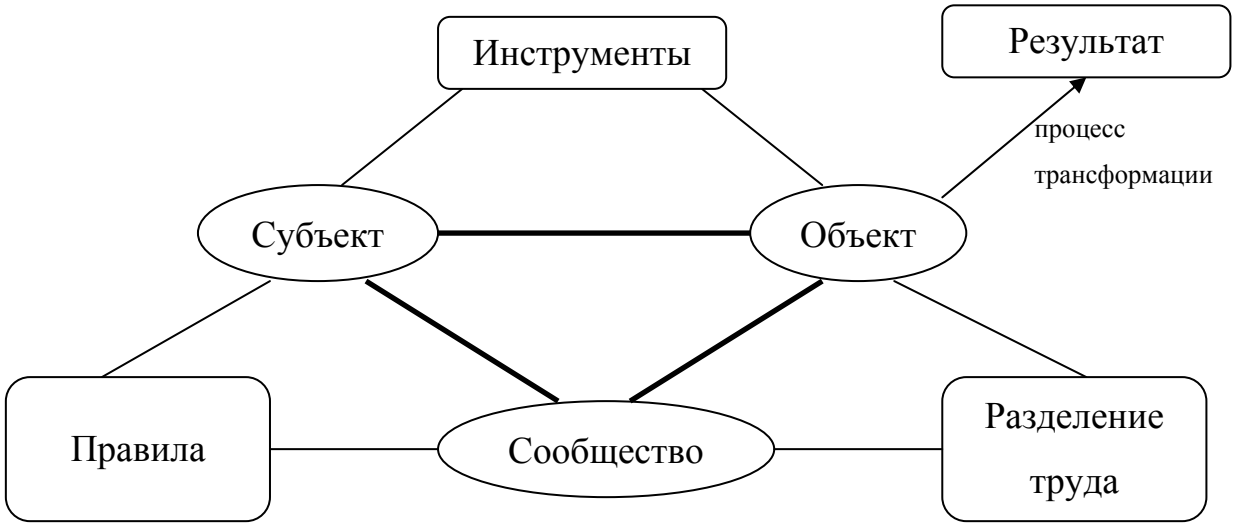

Рис. 1. «Треугольник Энгестрёма»: схема структуры деятельности, включающая основные (эллипсы) и опосредствующие (закруглённые прямоугольники) элементы деятельности, выделяемые в развитийном исследовании труда.

Каждая из этих деятельностей обладает собственным объектом и собственной структурой, которая также постоянно меняется в ходе развития трудовой практики. В рассматриваемых исследованиях очень часто используется модель структуры деятельности, предложенная Энгестрёмом [17] и называемая иногда «треугольником Энгестрёма» (см. рис. 1).

Эта модель показывает основные элементы деятельности и их связи между собой. Деятельность обладает не аддитивной, а системной структурой, и изменение одних её элементов неизбежно влияет на другие. В любой деятельности, по определению, существует субъект и объект, претерпевающий преобразования в результате этой деятельности. Отмечается, что англоязычный термин «activity» (который выносится в название теории — «activity theory») не полностью передаёт содержание русского слова «деятельность» или немецкого «Tätigkeit», так как последние подразумевают активность, обязательно направленную на преобразование какого-либо объекта [34].

В случае коллективной деятельности субъект является при этом частью сообщества и воздействует на объект совместно с ним. Каждая из связей между основными элементами деятельности является не прямой. Она опосредствуется некоторым артефактом (см. далее). Так, связь между субъектом и объектом опосредствуется инструментами, то есть всем, что используется в процессе трансформации, включая как физические, так и психические инструменты. Связь между субъектом и сообществом опосредствуется некоторыми эксплицитными и имплицитными правилами, то есть нормами и законами взаимодействия отдельных субъектов как членов сообщества. Связь между сообществом и объектом опосредствуется разделением труда, то есть эксплицитной и имплицитной организацией сообщества по отношению к объекту деятельности.

Каждая из этих составляющих деятельности не является статической. Очень много внимания в исследованиях в рамках HCI уделяется динамике развития деятельности, что вполне соответствует общефилософской методо-

логии, принятой в советской психологии. Инструменты, правила, разделение труда, а также и основные элементы деятельности постоянно развиваются и изменяются. Эти изменения можно рассматривать, как результат отдельных деятельностей. Таким образом, мы имеем для каждой деятельности также ряд «соседствующих» (neighboring): деятельность по преобразованию её инструментов, деятельность по разработке её правил и деятельность по развитию её субъекта [11, 17]. При этом стоит понимать, что каждая из деятельностей высшего уровня может подчинять себе несколько деятельностей первого уровня. Эти подчинённые деятельности мы будем называть «объектными» деятельностями (это четвёртый тип «соседствующей» деятельности). В рамках такой модели хорошо описывается, например, система уровней методологии. Исследования в рамках теории познания являются деятельностью по разработке инструментов и правил для разработчиков общенаучной методологии и т. д. Последним, пятым, типом «соседствующих» деятельностей является та же деятельность, но в исторически более продвинутом варианте. Такая деятельность может быть очень сходна по структуре с первоначальной, но имеет новый объект, а значит, и новый смысл для субъекта. Иллюстрацию подобной сети деятельностей см. на рисунке 2.

Между различными деятельностями могут возникать конфликты и противоречия. Например, между деятельностью по управлению финансами компании и деятельностью по разработке приложений неизбежно возникает противоречие, так как в рамках первой необходимо снижение затрат, что отрицательно влияет на результаты второй. Такие противоречия служат источником развития (что соответствует одному из основных законов диалектического материализма), вынуждая развиваться и ту, и другую деятельность [13, 33, 34]. Также возможны конфликты и внутри одной и той же деятельности. Ведь сообщество может состоять из субъектов нескольких категорий, которые обладают различными инструментами и правилами. Так, например, в рамках единой деятельности разработки приложения может возникнуть конфликт между программистами и разработчиками интерфейса.

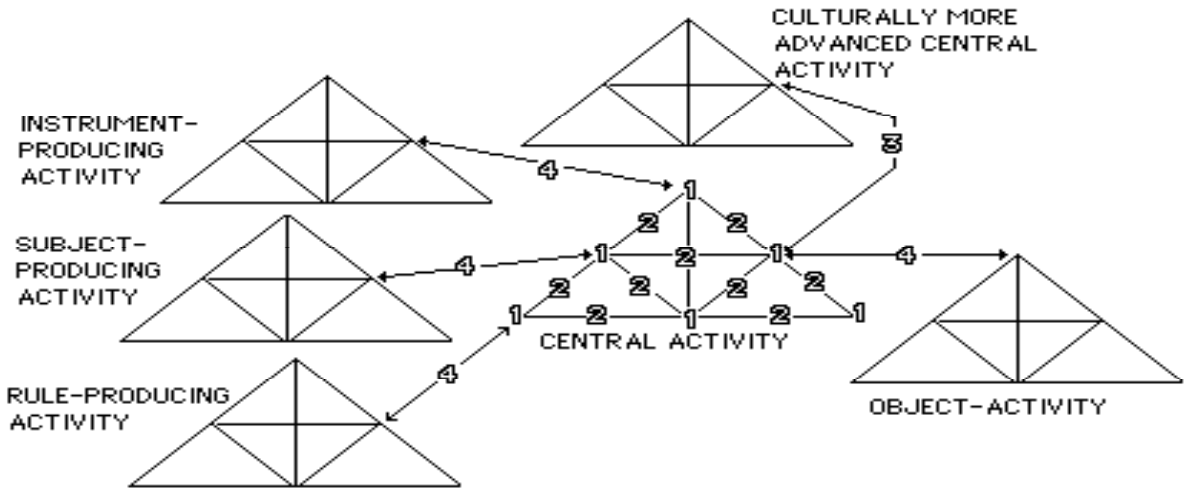

Рис. 2. Схема сети «соседствующих» деятельностей с обозначением первичных, вторичных, третичных и четверичных противоречий. По [17].

Энгестрём выделил 4 типа противоречий (см. рис. 2): первичные, вторичные, третичные и четвертичные [17]. Первичные противоречия — это противоречия в рамках одного элемента деятельности, например, внутри субъекта или между различными инструментами. Например, покупка более мощного аппаратного обеспечения повысит эффективность работы, но при этом может быть очень дорога. Это противоречие внутри инструмента «новое мощное аппаратное обеспечение» как элемента деятельности.

Вторичные противоречия — это противоречия между различными элементами одной и той же деятельности. Типичным примером является противоречие между жёстким разделением труда и новыми инструментами, поддерживающими взаимодействие субъектов на новом уровне. В таком случае первое не даёт раскрыться возможностям второго.

Третичные противоречия — это противоречия между объектами данной деятельности и исторически более продвинутой формы деятельности. Так, например, когда учителя и родители вынуждают ребёнка более серьёзно относиться к учёбе, то есть перейти от игровой мотивации к учебной, ребёнок может сопротивляться этому переходу к исторически более продвинутой деятельности, так как имеется противоречие между старым и новым объектом деятельности. В рамках концепции Д. Б. Эльконина, такое противоречие

разрешается через кризис. В HCI примером третичного противоречия может быть, например, введение объектно-ориентированных пользовательских интерфейсов [2]. Приказ начальства о переходе к новому, более продвинутому типу интерфейса может формально выполняться, и новые интерфейсы будут выглядеть «объектно-ориентированными», однако при этом идеология разработчиков и глубинное устройство системы не меняются. Возможно, чтобы произошли необходимые изменения, нужно добиться возникновения в сообществе некоторого рода кризиса.

Четвертичные противоречия — это противоречия между некоторой деятельностью и «соседствующими» с ней деятельностями (за исключением «соседствующей» деятельности пятого типа). Примером четвертичных противоречий являются противоречия между сообществом и новыми инструментами. Так, персонал, привыкший к старому программному обеспечению, может отказываться от нового, объективно более удобного и эффективного [3].

Основные понятия: артефакт

В зарубежной литературе широко используется такой термин, как артефакт. Здесь он подразумевает некоторый культурный предмет, причём он может быть как физическим, так и идеальным. Инструменты, схемы разделения труда, правила — это и есть артефакты. Сама программа, над которой трудятся разработчики или картинка, которую рисует пользователь в графическом редакторе, также являются артефактами.

Данное понятие по объёму значений практически соответствует понятию «орудие» в отечественной литературе. Отечественный термин «орудие» переводится в англоязычной литературе как «tool». Так же в рамках развитийного исследования труда называются те артефакты, которые опосредствуют взаимоотношения между субъектом и объектом. Однако в данной работе слово «tool» в последнем значении переводится, как «инструмент», чтобы избежать путаницы между понятиями. Важно и ещё одно различение: хотя слово «артефакт» может ассоциироваться у психологов с артефактами

в экспериментальных данных, значение, в котором оно используется в данной работе, является более широким. По определению, артефакт — это нечто, созданное человеком. В экспериментальных же данных артефакт — это ложные результаты, обусловленные ошибкой экспериментатора при формировании условий эксперимента.

Данное понятие позволяет адекватно описывать включённость внешних предметов в деятельность человека. Норман, указывавший на пренебрежительное отношение к роли вещей в психологии, выдвинул понятие «когнитивный артефакт» [43]. Затем, с приходом теории деятельности в HCI, это понятие было заменено аналогичным теоретико-деятельностным понятием. Значения этих понятий практически совпадают, однако последнее имеет более широкий систематический контекст, что позволяет делать определённые новые выводы.

Понятия артефактов и опосредствования позволяют расширить и переработать сам предмет HCI, что делает это название в известном смысле архаизмом. В рамках деятельностного подхода изучается не взаимодействие человека и компьютера, а человеческая деятельность (как индивидуальная, так и коллективная), опосредствованная компьютером [29].

Человеческая психика всегда определяется социокультурным окружением. А это окружение есть те опосредствующие элементы, те артефакты, которые участвуют в деятельности человека. В этих артефактах кристаллизовался опыт трудового сообщества [6]. Именно поэтому рассмотрение любой системы трудовой практики должно начинаться с изучения исторического контекста этой практики.

Основные понятия: структурные уровни деятельности

Иерархическая структура деятельности описывается в соответствии с классической схемой, предложенной А. Н. Леонтьевым [1]. Деятельность состоит из действий, которые производятся при помощи операций. В деятельности по разработке программы встречаются такие действия, как организация собраний, проведение юзабилити-тестирования, написание уча-

стка кода, отвечающего за отдельный модуль программы, и такие операции, как применение определённых команд языка программирования и печатание кода в редакторе.

Отмечается, что действия могут превращаться в операции, и обратно, при изменении условий, развитии навыка или в случае непредвиденных событий. Переключения, вызываемые непредвиденными событиями («деавтоматизации» в терминологии Н. А. Бернштейна) называются исследователями «сбоем» хода деятельности (breakdown). Однако так называемые «переключения фокуса» (focus shifts) внимания также могут и целенаправленно вызываться субъектом. Так происходит, например, в случае обучения другого некоторой операции. При этом она с необходимостью превращается в действие [11].

Также отмечается, что одно и то же действие может входить во многие деятельности, а одна и та же деятельность может осуществляться при помощи различных действий [6]. При этом одно и то же действие в рамках разных деятельностей обладает для субъекта различным личностным смыслом [34].

При рассмотрении уровней деятельности встаёт вопрос о сущности сознания. Ведь основное отличие операций от действий в их статусе по отношению к сознанию. Теория деятельности вносит в HCI вопрос о сознании, которому раньше уделялось не так много внимания [39].

Уровни взаимодействия в коллективном труде

Коллективный труд рассматривается как «субъект-объект-субъектные» связи [16]. Под этим подразумевается взаимодействие субъектов при работе с одним и тем же объектом. При его изучении выделяются три уровня взаимодействия: уровень координации, уровень кооперации и уровень сконструирования [6] (или уровень рефлексивной коммуникации, по терминологии Энгестрёма [16]).

Уровень координации действует при рутинной работе группы. При такой работе отдельные члены сообщества выполняют свою работу без эксплицитного взаимодействия с другими субъектами. Однако их

взаимодействие опосредствовано некоторыми правилами и алгоритмами работы, которые могут быть представлены как в форме официальных документов (например, технических заданий и спецификаций), так и в форме негласных правил работы в организации и в обществе в целом.

Уровень кооперации подразумевает взаимодействие субъектов, при котором они влияют на действия друг друга. При работе на этом уровне «субъекты сосредотачиваются на общей задаче, стараясь найти приемлемый для каждого из них способ концептуализировать и решить её» [16].

Со-конструктивный уровень — это уровень, на котором взаимодействие осуществляется с целью изменения самой рабочей практики: её инструментов и правил.

Некоторый уровень есть характеристика конкретного момента работы, однако они могут динамически переключаться между собой. Рассмотрим деятельность программистов. Обычно команда программистов работает на уровне координации. Каждый из программистов пишет свой участок кода или модуль программы. При этом они работают по заранее оговоренному плану в рамках одного языка и методологии программирования, имея некоторый образ результата своей деятельности, зная сроки сдачи работы и т. д. Прямое взаимодействие в данном случае не осуществляется.

Однако стоит программисту встретиться с каким-либо сложным или неоднозначным местом в спецификации проекта, он тут же свяжется с руководителем или со своими сотрудниками, чтобы решить, как стоит трактовать данный пункт спецификации. Это будет переходом на уровень кооперации.

Может случиться и так, что программисты со временем поймут, что, например, используемый язык программирования плохо подходит для данного проекта или сроки сдачи проекта слишком близки, и тогда они перейдут на со-конструктивный уровень взаимодействия, чтобы перестроить свою рабочую практику.

На уровне координации важно распределение труда. Переход на уровень кооперации активно задействует правила сообщества. Переход же к со-

конструктивному уровню переключает сообщество на «соседствующие» деятельности первых трёх типов и фактически является временной сменой объекта (а возможно, что и постоянной, в случае, если в ходе рефлексивного взаимодействия родится исторически более продвинутый вариант обсуждаемой деятельности), с которым работает данное сообщество.

Эта концепция в какой-то мере перекликается с концепцией сбоев хода деятельности. Однако последняя предназначена для изучения индивидуальной деятельности. Сбои также могут происходить на трёх уровнях взаимодействия: на уровне физического взаимодействия пользователя с компьютером (что вынуждает пользователя переключать внимание с объекта его основной деятельности на физические органы управления компьютером и, возможно, не позволяет переводить некоторые действия на уровень операций), на уровне аспектов управления системой (что также переводит фокус внимания на инструмент вместо объекта деятельности) и на уровне субъект-объектного взаимодействия [11].

Основные преимущества и ограничения теории деятельности

Исследователи HCI, работающие в рамках деятельностного подхода, выделяют такие преимущества этого подхода, как изучение естественных условий труда [39], строгость терминологии, многоуровневость, учёт контекста адекватного масштаба, рассмотрение деятельности в динамике и в развитии [34], большие возможности индукции, основанные на широте системы понятий, а также её «плотности» (большом количестве внутренних связей) [40].

Под многоуровневостью К. Кутти подразумевает то, что теория деятельности позволяет изучать физическое взаимодействие человека и компьютера, концептуальное взаимодействие и деятельность в определённом контексте в рамках одной методологии. В классическом же HCI физическое и техническое взаимодействие изучалось в рамках бихевиористского подхода (что соответствует уровню операций в теории деятельности), концептуальное

взаимодействие изучалось в рамках психологии обработки информации (уровень действий), а «контекстное взаимодействие» изучалось в рамках мультидисциплинарного подхода в виде «обогащённой психологии обработки информации» (enriched information processing psychology) (уровень деятельности).

Теория деятельности включает все те возможности, которые так долго обсуждались исследователями как столь необходимые для HCI: учёт особенностей субъекта, опосредствование, историчность, динамичность и т. д.

Рассмотрение деятельности в динамике наводит мысли на более подробную разработку поддержки со-конструктивного уровня взаимодействия. Ведь если деятельность развивается и существуют деятельности по изменению используемых в деятельности артефактов, значит, при разработке информационных систем стоит задуматься о поддержке этих деятельностей.

Очень важно, что теория деятельности рассматривает контекст адекватного масштаба, то есть не сужает анализ до уровня индивидуального взаимодействия с компьютером, но и не расширяет его излишне, что сделало бы анализ ещё менее осмысленным, чем при чрезмерном уменьшении его масштаба.

В. Н. Каптелинин указывает на несколько ограничений теории деятельности [30]. Во-первых, изначально теория деятельности была более приспособлена для описания индивидуальной деятельности. Хотя Энгестрём и некоторые другие исследователи значительно изменили такое положение дел, основываясь на понятии коллективного субъекта, всё-таки, по мнению Каптелинина, существуют некоторые проблемы с её использованием для изучения коллективной деятельности. Во-вторых, в рамках теории деятельности культура рассматривается несколько более узко, чем в культурно-историческом подходе Выготского. Ценности, мотивация, личность и т. д. изучаются в теории деятельности лишь в той мере, в которой они позволяют рационально объяснить взаимодействие человека с миром. В-третьих, понятие опосредствования, являющееся одним из основных преимуществ теории

деятельности, с другой стороны, также привносит некоторые ограничения. Так, в рамках нынешнего понятийного аппарата теории деятельности невозможно описать виртуальную реальность, так как она является артефактом, но при этом не опосредствует отношения с реальностью, а сама является реальностью особого рода. В-четвёртых, в рамках деятельностного подхода фактически не было готовых методик, применимых в области HCI, в отличие, например, от области образования.

Все эти проблемы решаемы и в какой-то степени уже решаются. Многие из них можно решить путём интеграции теории деятельности с некоторыми другими подходами.

Практическое применение

Информационные технологии позволяют поддерживать различные элементы деятельности на различных её структурных уровнях. Так, например, они поддерживают инструменты на уровне операций в форме автоматизации некоторых рутинных действий. Поддержка же инструментов на уровне деятельности состоит в возможности создания новых инструментов и автоматизации других действий. Также информационные системы позволяют производить манипуляционные действия с объектами, визуализировать объекты и действия, делают некоторый объект общим, вынуждают следовать выработанным правилам и позволяют вырабатывать новые, позволяют субъектам взаимодействовать между собой и создавать новые сообщества. В рамках классических исследований HCI наиболее хорошо развивались системы обеспечения деятельности на уровне операций. Целью современного HCI является разработка систем, поддерживающих уровень кооперации и соконструктивный уровень взаимодействия.

Рассмотрение процессуально-ориентированных сред разработки программного обеспечения с позиций теории деятельности

Процессуально-ориентированные среды разработки программного обеспечения (Process-Centered Software Development Environments, далее

PCSDEs) [6] — это программы поддержки работы команды разработчиков, которые помогают планировать работу, собирают различные документы, относящиеся к данному проекту, в некую компактную структуру, запускают различные редакторы и другие манипуляторные приложения.

В PCSDEs существует терминология, имеющая частичное внешнее сходство с терминологией теории деятельности. Однако значения многих терминов не соответствуют друг другу. Например, «деятельностью» в терминологии PCSDEs называют просто некоторую ступень проекта. PCSDEs следят за тем, какие «деятельности» (в указанном выше значении) на данный момент актуальны и предоставляют необходимую информацию. По мере их выполнения одни деятельности сменяются другими.

Теоретико-деятельностный анализ позволяет достаточно легко выявить недостатки и преимущества этих сред. Дело в том, что они, как видно, отлично поддерживают работу на уровне координации. Однако с уровнями кооперации и со-конструирования существует больше проблем.

На уровне кооперации от среды требуется поддержка взаимодействия, как эксплицитного (в форме видеоконференций, форумов и т. п.), так и имплицитного (в виде поддержки одновременной работы над одним документом). Эксплицитное взаимодействие часто осуществляется независимо от среды: по телефону или с использованием программ мгновенной передачи сообщений (таких, как ICQ™). Достаточно редко в среду бывают встроены CSCW-модули. Практически отсутствуют системы автоматического протоколирования взаимодействий. А если субъектам необходимо каждый раз самостоятельно публиковать результаты взаимодействия, это значит, что среда не выполняет своих функций по автоматизации. Имплицитное взаимодействие также является важной проблемой. Ведь одновременная работа с одним документом может вызвать то, что результаты работы одного человека будут записаны поверх изменений, произведённых другим. Некоторые среды просто блокируют используемый кем-либо документ для других пользователей. Однако иногда одновременная работа нескольких субъектов над одним до-

кументом совершенно необходима. Для этого важно разработать соответствующую систему кооперированной трансформации документа. Также желательно, чтобы эксплицитное и имплицитное взаимодействие могли быть интегрированы друг с другом.

Поддержка со-конструктивного уровня взаимодействия разработана в имеющихся на данный момент средах ещё меньше. Обычно для изменения структуры процесса необходимо перепрограммировать эти приложения. Зачастую это невозможно по техническим причинам, так как эти среды обычно представляют собой скомпилированные приложения, предоставляемые сторонними разработчиками. Изредка встречаются довольно сложные встроенные инструменты, позволяющие что-то менять в «процессуальной модели» проекта, но они очень слабо интегрированы с поддержкой взаимодействия.

Таким образом, видно, что для развития сред необходимо в первую очередь улучшать поддержку верхних уровней взаимодействия. Такой анализ оказался возможен благодаря понятию об уровнях взаимодействия, которое родилось в рамках деятельностного подхода. Отсутствие подобной теоретической базы крайне затруднило бы анализ и, скорее всего, сделало бы его результаты более фрагментарными.

Интересная форма поддержки со-конструктивной деятельности архитекторов была предложена в группой швейцарских исследователей [20]. Система BUILD-IT — это система изменённой реальности. Она позволяет архитекторам, сидя за столом, обсуждать свои идеи насчёт устройства, например, парка или городского квартала. Для обозначения различных объектов они пользуются настоящими, физическими кубиками, расставляя их на столе. Также они могут рисовать что-либо на столе и т. д. Благодаря современной системе видеорегистрации, а также встроенным в кубики передатчикам, компьютер способен переносить построенный на столе квартал в трёхмерную компьютерную графику и отображать итоговую картинку в режиме реального времени.

Идеалом развития сред разработки программного обеспечения будет появление подобных инструментов, позволяющих формировать трудовую практику в режиме постоянного взаимодействия и с возможностью максимально наглядного и достоверного прогнозирования результатов обсуждаемых изменений.

Разбор программ по подготовке презентаций

В качестве другого примера удачной концептуализации при помощи деятельностного подхода можно привести исследование программ, используемых для подготовки презентаций [40]. Б. Нарди выделила два класса приложений: общие и специальные. Среди общих приложений встречаются такие, как текстовые редакторы, графические редакторы и т. д. Программы для создания презентаций (например, Microsoft PowerPoint™) являются специальными программами. Изучив 16 человек, часть из которых профессионально занимаются созданием презентаций, а часть создают их лишь время от времени, методом интервью (который в HCI обычно характеризуется, как один из этнологических методов), Нарди обнаружила, что многие пользователи предпочитают общие приложения, несмотря на то, что специфическое приложение делает работу над презентацией более автоматизированной. Например, для изменения размера слайда в общих приложениях придётся вручную менять размер каждого из них.

Те, кто составляют презентации лишь изредка, просто не желают тратить деньги и время на покупку и освоение специфической программы. Однако почему профессионалы предпочитают общие приложения?

Как стало ясно из интервью, специализированные приложения используются ими для создания простых, черновых презентаций. В случае необходимости сложной графики или вёрстки большого объёма текста они пользуются общими приложениями.

Было ясно, что существуют два вида специфичности: специфичность подзадачи и специфичность задачи высшего уровня. Программы, специфичные для подзадачи, позволяют более удачно выполнять эту конкретную под-

задачу, а программы, специфичные для задачи высшего уровня поддерживают её выполнение от начала и до конца.

Сначала исследователи ввели такие неуклюжие термины, как «подзадача» и «задача высшего уровня», однако затем осознали, что данный случай гораздо более легко и систематично решается при помощи терминологии теории деятельности. Так, подзадача — это, на самом деле, некоторое действие, которое может быть использовано в нескольких деятельностях. Задача высшего уровня — это целая деятельность, состоящая из нескольких или многих действий.

Теперь легко увидеть, как должны соотноситься между собой общие и специфичные приложения (см. рис. 3). Специфичные приложения должны интегрировать и синхронизировать работу общих. Эту задачу можно выполнить так, чтобы специфичные приложения как бы не существовали, а существовала лишь взаимная интеграция общих. Такой подход к созданию информационных систем получил название «инструментального».

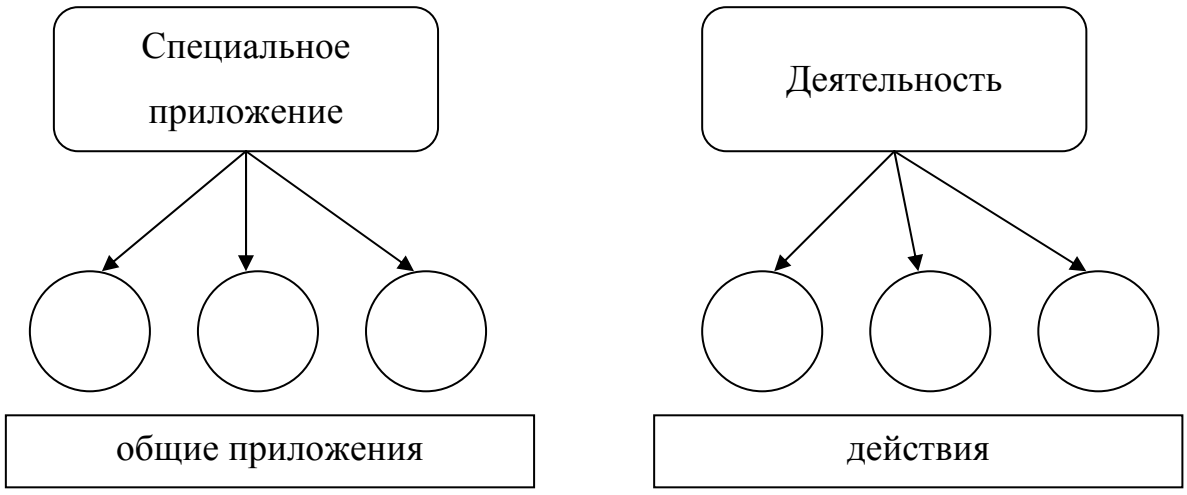

Рис. 3. Соотношение видов приложений и структурных единиц деятельности

Системно-структурный анализ деятельности

Методика системно-структурного анализа трудовой деятельности — это структурированная форма анализа некоторой формы труда, включающая как качественный, так и количественный анализ и основанная на методологии теории деятельности. Она развивается в трудах Г. З. Бедного и его кол-

лег [7, 8]. Эта методика в основном приспособлена к изучению индивидуальной деятельности.

Первичный качественный анализ деятельности подразумевает описание общего её хода, её смысла и цели. В первичном качественном анализе описываются условия деятельности, имеющиеся инструменты деятельности и т. д. Обязательно описывается субъект данной деятельности: требования, предъявляемые к личностным, трудовым и физическим свойствам такого человека. Также описываются возможные индивидуальные стили деятельности [9]. Понятие индивидуальных стилей деятельности, возможно, способно помочь в развитии гибкости систем. Российскими исследователями наблюдались индивидуальные различия в стратегиях поведения испытуемого пользователя в сложных условиях отвлекающего прерывания [12]. Вероятно, эти различия можно объяснить формированием индивидуального стиля деятельности, компенсирующего такие качества субъекта, как, например, рассеянность.

После этого производится дальнейший качественный анализ, в рамках которого используется алгоритмический анализ деятельности. Вся деятельность подразделяется в ходе алгоритмического анализа на некоторые элементы. Эти элементы делятся на операторы и логические условия. Первые включают одно или несколько действий, вторые определяют порядок операторов. Для записи таких алгоритмов существует особая нотация [7], которая позволяет отмечать типы каждого из этих элементов и их взаимосвязи.

Алгоритмический анализ используется также и в когнитивном подходе в таких моделях, как GOMS [7, 30]. Однако данный подход соотносит единицы анализа со строго определёнными понятиями теории деятельности, а также позволяет описывать вероятностные алгоритмы.

Проведение алгоритмического анализа позволяет по-новому посмотреть на первичные качественные данные. Этот процесс имеет циклический характер. Постепенно продвигаясь в анализе, мы можем находить всё новые и новые недостатки изначальной системы труда.

Затем на базе имеющегося алгоритмического описания мы можем составить временную характеристику деятельности, изучив длительность различных её частей. При помощи этого анализа мы можем количественно оценить эффективность изменений в ходе деятельности.

Важно понимать, что системно-структурный анализ деятельности имеет смысл только тогда, когда он проводится как целое. Нельзя обойтись только первичным качественным или только алгоритмическим анализом.

Анализ видеозаписей

В исследовании информационных систем часто используются видеозаписи взаимодействия человека и компьютера. Теория деятельности предоставляет несколько новых методик анализа видеоинформации. Одной из методик является рассмотренный выше алгоритмический анализ как часть системно-структурного анализа деятельности.

Другая методика основана на понятии о сбоях деятельности [11]. Во взаимодействии человека с компьютером выделяются различные имеющиеся точки фокуса внимания. Одни из этих точек относятся к уровню объект-субъектного взаимодействия, другие — к уровню управления системой. В таких случаях подробно записываются все переключения внимания субъекта. При анализе этих переключений обязательно надо выяснить, откуда и куда переключилось внимание пользователя, было ли это намеренным переключением или сбоем и какова причина сбоя.

Анализ противоречий

При изучении групповой деятельности для нас более важным оказывается анализ противоречий [13]. Во всякой деятельности можно выделить какие-либо противоречия, принадлежащие к 4 типам, описанным Энгестрёмом [17]. Для повышения эффективности деятельности рассматриваются и, по возможности, разрешаются все найденные противоречия. С этой целью обычно выделяются и исследуются различные группы субъектов (например, программисты, художники, менеджеры, пользователи), имеющие какое-либо отношение к данной деятельности. Для этого интервьюируют представителей

каждой из этих групп, проводят этнологические наблюдения. После того, как конфликт обнаружен, он может решаться, например, при помощи организации общего семинара, где имеются представители каждой группы субъектов.

Выводы

Теория деятельности открывает новые просторы для исследований в рамках HCI, несколько изменяя и расширяя предмет дисциплины. Несмотря на то, что некоторые исследователи считают теорию деятельности чересчур трудной для усвоения и недостаточно полезной на практике, многими исследователями было показано, что эти выводы поспешны, и теория деятельности является полезным инструментом в разработке многих систем.

Теория деятельности показывает себя особенно эффективной в таких дисциплинах, как CSCW и ПО-инженерия, то есть в изучении совместной деятельности людей. Индивидуальное взаимодействие с компьютером на техническом уровне — это та область, в которой классические когнитивные исследования всё-таки достигли достаточно высокого уровня и позволяют разрабатывать весьма удобные интерфейсы.

Гораздо более ответственным является вопрос о том, как сделать систему полезной. Ведь человек всё-таки достаточно адаптивное существо, чтобы приспособиться к самой неудобной системе (хотя это не значит, что их не стоит делать удобными), но только в том случае, если она способна удовлетворить его потребности и если она удачно поддерживает и опосредствует ту деятельность, которой он занимается.

Теория деятельности позволяет формализовать и достаточно однозначно описать то, что считалось недоступным для объективного описания. Описание труда в деятельностном подходе решает проблемы, которые имелись как в когнитивном, так и чисто этнографическом подходе. Она объединяет в себе идеографическое и номотетическое описание, не теряя из виду социальные аспекты труда и при этом описывая их достаточно строго.

Работа с информационными системами в рамках деятельностного подхода также включает в себя этнографические и экспериментальные методы. Однако теория деятельности при этом играет роль методологии более высокого уровня, позволяя объединить и концептуализировать данные, полученные при применении этих методик.

Описывая трудовую практику в рамках теоретико-деятельностной терминологии, мы можем прояснить себе все проблемы данной системы и только в этом случае мы можем разрабатывать, точно зная, что мы разрабатываем и каких целей надо добиться. По увеличившемуся в последние годы количеству журналов и конференций видно, что приложение теории деятельности к разработке информационных систем активно развивается и будет развиваться в дальнейшем. А появление компаний, которые заявляют, что их работа в первую очередь основывается на разработках деятельностного подхода, показывает, насколько широк диапазон его применений внутри HCI.

Анализ работ по приложению теоретико-деятельностного подхода к разработке информационных систем позволяет более строго переформулировать данное во введении определение информационной системы как *совокупности артефактов, опосредствующих связи в сети деятельностей, осуществляющей приём, хранение, преобразование, изъятие и использование информации в рамках рабочей практики некоторого индивида или социальной группы.* Вместо понятия «орудие» предлагается использовать понятие «артефакт», имеющее более широкое поле значений в естественном языке и позволяющее избежать путаницы между орудиями и инструментами. Расширенный предмет HCI составляет *деятельность индивидуального или коллективного субъекта, опосредствованная компьютером как частью информационной системы*.

Выше был дан неполный обзор практических методик, предложенных в рамках деятельностного подхода к HCI. Целью данной работы являлось не подробное описание процедур этих методик, а отражение возможностей деятельностного подхода в данной области знаний.

Современные исследования постепенно сводят на нет обнаруженные ограничения теории деятельности. Теория деятельности становится в большой степени ориентированной на коллективную деятельность, появляются новые практические методики, теория деятельности объединяется с некоторыми более частными методологиями, таким образом, она сама многое приобретает от своего «сотрудничества» с HCI. По-видимому, следующим рубежом, который предстоит перейти сторонникам этого подхода, будет создание фундаментального руководства по HCI, объединяющего имеющиеся знания в рамках единой теоретической основы, а также описание всего цикла создания программного обеспечения на основе того же подхода.

Количество практических методик особенно быстро растёт в последние годы [14, 45]. В основном, они анализируют различные аспекты деятельности и направлены на выявление «проблемных мест» и интерпретацию причин имеющихся сложностей. Они могут объединиться на основе системно-структурного анализа деятельности и стать основой теоретико-деятельностного цикла разработки информационной системы. Он может быть представлен как итеративный цикл со сменой фаз, представленных разными уровнями взаимодействия. Концептуальная разработка продукта и составление требований производится на уровне кооперации, а возможно, и соконструирования, построение интерфейса — на уровне кооперации, а реализация интерфейса — на уровне координации. Естественно, на каждом из этапов могут происходить переключения на другой уровень. На каждом этапе необходимо использовать свои методы анализа. Так, при разработке требований понадобится, например, анализ противоречий, при построении интерфейса — алгоритмический анализ, а на этапе тестирования — количественно-временной анализ.

Список литературы

1. Леонтьев А. Н. *Деятельность. Сознание. Личность*.— М.: Смысл; Издательский центр «Академия», 2004.
2. Мандел Т. *Разработка пользовательского интерфейса*.— М.: ДМК Пресс, 2001.
3. Сатин Д. *Спонтанно формирующиеся бизнес-стандарты пользовательского интерфейса* (Устное выступление на первом семинаре RusCHI. Москва, 18 декабря 2004. Презентация PowerPoint™: http://www.sigchi.ru/Seminars/01/Satin.ppt)
4. Bannon L. J. A pilgrim's process: From cognitive science to cooperative design // *AI and Society*, 1990, 4.
5. Bannon L. J. From human factors to human actors: The role of psychology and human-computer interaction studies in system design // *Design at Work: Cooperative Design of Computer Systems* / J. Greenbaum, M. Kyng (eds.).— Hillsdale, NJ: Lawrence Erlbaum, 1991.
6. Barthelmess P., Anderson K. M. A view of Software Development Environments based on Activity Theory // *Computer-Supported Cooperative Work,* 11, 2002.
7. Bedny G. Z., Karwowski W. A Systemic-Structural Activity Approach to the Design of Human-Computer Interaction Tasks // *International Journal of Human-Computer Interaction*, Vol. 16, No. 2, 2003.
8. Bedny G. Z., Karwowski W., Kwon Y.-G. A methodology for systemic-structural analysis and design of manual-based manufacturing operations // *Human Factors and Ergonomics in Manufacturing*, 11, 2001.
9. Bedny G. Z., Seglin M. H. Individual style of activity and adaptation to standard performance requirements // *Human Performance*, 1, 1999.
10. Bertelsen O. W., Bødker S. Introduction: Information Technology in Human Activity // *Scandinavian Journal of Information Systems*, 12, 2000.
11. Bødker S. Applying Activity Theory to Video Analysis: How to Make Sense of Video Data in HCI // *Context and Consciousness: Activity Theory and Human-Computer Interaction* / B. Nardi (ed.).— Cambridge, MA: MIT Press, 1996
12. Burmistrov I., Leonova A. Effects of Interruptions on the Computerised Clerical Task Performance // *Human-Computer Interaction: Human Aspects of Business Computing (Proceedings of East-West HCI'96)* / Overmyer S., Gornostaev J. (eds.).— Moscow: ICSTI, 1996.
13. Collins P., Shukla S., Redmiles D. Activity Theory and System Design: A View from the Trenches // *Computer-Supported Cooperative Work,* 11, 2002.
14. *Computer-Supported Cooperative Work,* 11, 2002.
15. *Context and Consciousness: Activity Theory and Human-Computer Interaction* / B. Nardi (ed.).— Cambridge, MA: MIT Press, 1996

16. Engeström Y. Coordination, Cooperation and Communication in the Courts // *Mind, Culture and Activity*.— Cambridge, MA: Cambridge University Press, 1997.
17. Engeström Y. *Learning by Expanding*. *An Activity-theoretical Approach to Developmental Research*.— Helsinki: Orienta-konsultit Oy, 1987.
18. Engeström Y., Miettinen R. Introduction to *Perspectives on Activity Theory* / Y. Engeström, R. Miettinen, R.-L. Punamäki (eds.).— Cambridge, MA: Cambridge University Press, 1999.
19. Engeström Y., Virkkunen J., Helle M., Pihlaja J., Poikela R. Change Laboratory as a tool for transforming work // *Lifelong Learning in Europe*, Vol. 1, No. 2, 1996.
20. Fjeld M, Lauche K., Bichsel M., Voorhorst F., Krueger H., Rauterberg M. Physical and Virtual Tools: Activity Theory Applied to the Design of Groupware // *Computer-Supported Cooperative Work,* 11, 2002.
21. Gal'perin P. Ya. The problem of activity in Soviet psychology. Stage-by-stage-formation as a method of psychological investigation // *Journal of Russian and East European Psychology* 30 (4), 1992.
22. Glaser B., Strauss A. *The Discovery of Grounded Theory*.— New York: Aldine Publishing, 1967.
23. Grudin J. CSCW: History and Focus // *IEEE Computer*, V. 27, No. 5, 1994
24. Hutchins E., Hollan J., Norman D. A. Direct manipulation interfaces // *User Centered System Design: New Perspectives on Human-Computer Interaction* / D. A. Norman, S. Draper (eds.).— Hillsdale, NJ: Lawrence Erlbaum, 1986.
25. Information system // Интернет-энциклопедия *Wikipedia* (web-адрес: http://en.wikipedia.org/wiki/Information_system), последнее обновление страницы в 2005 г.
26. *Information Systems & Activity Theory: Tools in Context* / H. Hasan, E. Gould, P. Hyland (eds.).— Wollongong: University of Wollongong Press, 1998.
27. *Information Systems & Activity Theory: Volume 2, Theory and Practice* / H. Hasan, E. Gould, P. Larkin, L. Vrazalic (eds.).— Wollongong: University of Wollongong Press, 2001.
28. *Information Systems & Activity Theory: Volume 3, Expanding the Horizon* / H. Hasan, I. Verenikina, E. Gould (eds.).— Wollongong: University of Wollongong Press, 2003.
29. Kaptelinin V. Activity Theory: Implications for Human-Computer Interaction // *Context and Consciousness: Activity Theory and Human-Computer Interaction* / B. Nardi (ed.).— Cambridge, MA: MIT Press, 1996
30. Kaptelinin V. Computer-Mediated Activity: Functional Organs in Social and Developmental Contexts // *Context and Consciousness: Activity Theory and Human-Computer Interaction* / B. Nardi (ed.).— Cambridge, MA: MIT Press, 1996

31. Koisitnen K. *Learning to Survive*: Ph. D. dissertation. Helsinki: Centre for Developmental Work Research and Activity Theory.
32. Kolari P. Applying activity theory in developing a framework for understanding and guiding system development in product development context // *Proceedings of the 8th European Conference of Computer-Supported Cooperative Work*, Helsinki, Finland, 14th September 2003.
33. Korpela M., Mursu A., Soriyan H. A. Information Systems Development as an Activity // *Computer-Supported Cooperative Work,* 11, 2002.
34. Kuutti K. Activity Theory as a Potential Framework for Human-Computer Interaction Research // *Context and Consciousness: Activity Theory and Human-Computer Interaction* / B. Nardi (ed.).— Cambridge, MA: MIT Press, 1996.
35. Latour B. *Science in Action: How to Follow Scientists and Engineers through Society*.— Cambridge, MA: MIT Press, 1987.
36. Leont'ev A. N. The problem of activity in psychology // *Soviet Psychology* 13 (2), 1974.
37. Leont'ev A. N. The problem of activity in the history of Soviet psychology // *Soviet Psychology* 27 (1), 1989.
38. Monk A., Nardi B., Gilbert N, Mantei M., McCarthy J. Mixing oil and water? Ethnography versus experimental psychology in the study of computer-mediated communication // *Proceedings INTERCHI'93 Conference on Human Factors in Computing Systems*. Amsterdam, April 24—29, 1993.
39. Nardi B. Activity Theory and Human-Computer Interaction // *Context and Consciousness: Activity Theory and Human-Computer Interaction* / B. Nardi (ed.).— Cambridge, MA: MIT Press, 1996.
40. Nardi B. Some Reflections on the Application of Activity Theory // *Context and Consciousness: Activity Theory and Human-Computer Interaction* / B. Nardi (ed.).— Cambridge, MA: MIT Press, 1996.
41. Neisser U. *Cognition and Reality*.— San Francisco: Freeman, 1976. (Или Найссер У. *Познание и реальность*.— М.: Прогресс, 1981.)
42. Nielsen-Norman Group. Сборники руководящих принципов (guidelines) по веб-дизайну — http://www.nngroup.com/reports/
43. Norman D. A. Cognitive artifacts // *Designing Interaction: Psychology at the Human-Computer Interface* / J. Carrol (ed.).— New York: Cambridge University Press, 1991.
44. Norman D. A. *The Design of Everyday Things*.— New York: Basic Books, 1988.
45. *Scandinavian Journal of Information Systems*, 12, 2000.
46. Whiteside J., Wixon D. Discussion: Improving human computer interaction — a quest for cognitive science // *Interfacing Thought: Cognitive Aspects of Human-Computer Interaction* / J. Carrol (ed.).— Cambridge, MA: MIT Press, 1987.