Санкт-Петербургский государственный университет

**Моторин Анатолий Евгеньевич**

**Выпускная квалификационная работа**

**Применение понятий теории медиа в исследовании электронной литературы (на примере текстов Мэри Энн Бриз)**

Уровень образования: магистратура

Направление 45.04.01 «Филология»

Основная образовательная программа ВМ.5612. «Литература и культура народов зарубежных стран»

Профиль «Литература народов зарубежных стран»

Научный руководитель:

доцент,

Кафедра истории

зарубежных литератур, Миролюбова Анастасия

Юрьевна

Рецензент: доцент, Департамент филологии, Санкт-Петербургский филиал ФГАОУВО «НИУ «Высшая школа экономики»,

Делазари Иван Андреевич

Санкт-Петербург

2021

**Оглавление:**

[Введение 3](#_Toc72688134)

[**Глава 1. Концептуализация современных технологий в теории медиа и электронной литературе** 8](#_Toc72688135)

[1.1 Теории медиа 8](#_Toc72688136)

[1.1.2. Информация и шум 13](#_Toc72688137)

[1.1.3 Исполнение 16](#_Toc72688138)

[1.1.4. Программный код 20](#_Toc72688139)

[1.1.5. Знак 24](#_Toc72688140)

[1.1.6. Память 27](#_Toc72688141)

[1.1.7. Коммуникация 29](#_Toc72688142)

[1.1.8. Сети и протоколы 32](#_Toc72688143)

[1.2. Софтвер 33](#_Toc72688144)

[1.3. Электронная литература 37](#_Toc72688145)

[1.3.1. Понятие э. литературы и ее особенности как отдельного медиума 37](#_Toc72688146)

[1.3.2. Code work 42](#_Toc72688147)

[Выводы к первой главе 44](#_Toc72688148)

[**Глава 2. Практика Мэри Энн Бриз в медиатехнической среде 45**](#_Toc72688149)

[2.1. Художественная и литературная практики Мэри Энн Бриз 45](#_Toc72688150)

[2.1.1 Устоявшиеся интерпретации текстов mez 46](#_Toc72688151)

[2.2 Рецепция текстов mez в современном медиатехническом окружении. Критика интерпретации текстов mez как языка-пиджина 53](#_Toc72688152)

[2.2.1 Применение понятия интерактивности в интерпретации электронной литературы и текстов mez. «User amplification» как основное условие взаимодействия человека и компьютера 53](#_Toc72688153)

[2.2.2 Проблема границ чтения и письма в текстах mez. 56](#_Toc72688154)

[2.2.3 Применение терминов теории систем и кибернетики второго порядка к текстам mez 60](#_Toc72688155)

[2.2.4 Системы амплификации и системы означивания 62](#_Toc72688156)

[2.2.5 Амплификация и «остранение» 72](#_Toc72688157)

[2.3 Исполнение и выполнение 74](#_Toc72688158)

[Заключение 79](#_Toc72688159)

[Список использованной литературы: 82](#_Toc72688160)

# **Введение**

Данная выпускная квалификационная работа посвящена анализу работ австралийской художницы и писательницы Мэри Энн Бриз, пишущей под псевдонимом mez. Первые работы Мэри Энн Бриз появились в 1993 году, когда она начала использовать Интернет для создания и распространения своих текстов. На сегодняшний день она является автором множества работ в жанре цифрового искусства: VR-инсталляций, интерактивных нарративов, экспериментальных компьютерных игр и других произведений. Центральным для данного исследование будет ее проект Mezangelle – язык, состоящий из элементов естественных и формализованных языков.

Ее практика вписывается в художественное направление под названием net.art, появившееся в 1990-ых годах и объединявшее писателей и художников, осмыслявших зарождающуюся цифровую культуру, ее влияние на уже существующие виды искусства, их производство и дистрибуцию. Среди них были как художники, работавшие напрямую с новыми технологиями, создававшие программы, веб-страницы или компьютерные игры, осмысляющие современную им технокультуру (JODI, Ольга Лялина, I/O/D, Алексей Шульгин, The Bureau of Inverse Technology, Тед Уорнелл, Талан Меммот, Джон Кэйли и др.), так и писатели, которые с помощью существующих технологий расширяли возможности литературных и художественных форм: Алан Сондхейм, Майкл Джойс, Шелли Джексон, Брайан Леннон и др. Границы между двумя выше обозначенными группами авторов довольно подвижны, поскольку в цифровой среде существуют как инструменты для создания объектов современного искусства, так и для написания различных текстов.

В контексте появления новых технологий и их осмысления в современном искусстве и литературе практика Мэри Энн Бриз выделяется работой по осмыслению материальности кода, скрытых механизмов работы сетевых протоколов и языков программирования. Через свои тексты и их распространение с помощью электронной почты она проблематизирует существование литературного и любого другого текста в современном медиатехническом окружении, меняющиеся конвенции восприятия литературы и ее взаимодействие с другими медиумами.

**Актуальность** данного исследования обусловлена возникновением новых теоретических рамок для исследования литературы, ее трансформирующимися формами, не всегда аналитически схватываемыми уже существующими литературоведческими теориями. Результаты данной работы позволяют дать иной взгляд не только на работы Мэри Энн Бриз, но и на другие работы нет-арта, изучить то, как методы создания произведений электронной литературы повлияли на современную литературу и современное искусство, а также на организацию существующего пользовательского опыта и на возможные новые виды этой организации. **Научная новизна** заключается в ином подходе к текстам Мэри Энн Бриз, сосредоточенном не на содержании и формальной организации ее текстов, а на специфичных характеристиках программного обеспечения, с помощью которых создаются тексты, а также культурных и психологических импликаций этих технологий, влияющих на производство текстов.

**Цель** данного исследования заключается в использовании концептуальной рамки теории медиа для интерпретации текстов Мэри Энн Бриз. Данная цель предполагает решение следующих **задач**:

1. дать краткую характеристику медиа теории, объектам ее исследования и методологии.
2. описать основные способы концептуализации медиа, существующие на данный момент и разработанные разными авторами, дать определение основным понятиям теории медиа.
3. дать определение электронной литературе, описать основные жанры и провести разграничения между ней и существующими литературными практиками.
4. охарактеризовать существующие интерпретации текстов Мэри Энн Бриз.
5. дать иную интерпретацию текстам Мэри Энн Бриз, собранным в сборнике «Human Readable Messages», опираясь на методологию медиа-теории.

**Объектом исследования** являются тексты Мэри Энн Бриз, собранные в книге «Human Readable Messages». **Предметом** исследования являются способы означивания, существующие в этих текстах, а также медиатехнические условия, воздействующие на рецепцию этих текстов.

**Методологической основой** исследования послужил медиа-специфический анализ, описанный в работах К. Хэйлс и других медиатеоретиков, а также герменевтический метод для анализа отдельных аспектов текстов.

В настоящее время электронная литература достаточно представлена в отечественной гуманитарной науке. В крупных литературных журналах («Новое литературное обозрение», «Иностранная литература») выходили статьи и рецензии, посвященные этой теме. Среди работ на русском языке можно выделить монографию С. А. Кучиной «Электронный художественный текст: понятие, категории, лингвосемиотическая специфика», в которой представлен обзор отечественных и зарубежных работ по тематике, статьи Натальи Федоровой и Михаила Визеля. Медиа теория также изучается отечественными исследователями: «Исследовательский центр медиафилософии» выпускает статьи и проводит конференции, посвященные современной медиареальности.

В данном исследовании использовались теоретические положения, изложенные в текстах К. Хэйлс, С. Реттберга, В. Чан, Л. Мановича, М. Марино, Д. Тэнона, и др. Скотт Реттберг в своей книге «Электронная литература» описывает становление электронной литературы как совокупности практик художественной работы с текстом при помощи современных технологий, и как институции со своими способами теоретического осмысления произведений, способов сохранения и распространения текстов.

Особое внимание к технически опосредованным аспектам жизни уделяется в современной медиатеории, утверждающей опосредованность любой человеческой деятельности и экстернализацию важнейших когнитивных способностей в качестве ключевого фактора развития человека. В работе представлены труды таких медиатеоретиков, как Б. Стиглер, М. Хансен, Ф. Киттлер, Л. Манович, Ю. Хуэй, Ф. Крамер, М. Фуллер, В. Флюссер, Дж. Кокс. В качестве работ, осмысляющих коммуникацию, ее культурные и материально-технические аспекты, были использованы труды А. Гэллоуэя, Ю. Такера, М. Уорка.

Теоретической основой для медиа-специфического анализа послужили труды Б. Кларка, в которых он применяет теорию систем, сложившуюся в междисциплинарных исследованиях на стыке социологии, биологии и кибернетики, для интерпретации культурных артефактов, в частности повествовательных текстов. Для более точного анализа специфичных свойств программных сред, в которых производятся и распространяются тексты Мэри Энн Бриз, в работе используется концептуализация софтвера, описанная в книге Л. Мановича «Софтвер берет на себя командование».

При рассмотрении электронной литературы как отдельной совокупности практик, находящихся на пересечении литературы и современного искусства, были использованы работы таких теоретиков и практиков, как Т. Меммот, Д. Кэйли, С. Реттберг, К. Хэйлс, П. Уайлдмен, Р. Рэйли, А. Шульгин, А. Сондхейм.

Данная работа состоит из введения, двух глав, заключения и библиографического списка. Объем работы составляет 89 страниц. Во введении формулируется цель и задачи работы, а также краткий обзор теоретической базы исследования. Первая глава «Концептуализация современных технологий в теории медиа и электронной литературе» представляет собой обзор существующих на данный момент медиа теорий, описание основных терминов, с помощью которых осуществляется интерпретация современного медиатехнического окружения, и краткую характеристику электронной литературы, ее основных жанров и характерных особенностей.

Вторая глава «Практика Мэри Энн Бриз в медиатехнической среде» представляет собой попытку с помощью терминов теории систем и кибернетики второго исследовать методы производства текстов, связанные с программным кодом и взаимодействием пользователя с графическим интерфейсом, и системам означивания, во взаимодействии которых появляются литературные тексты. После этого мы рассмотрим, как с помощью текстов mez изменяются отношения между откладыванием значения и амплификацией пользователя. В последней части второй главы будут рассмотрены понятия «эксплойта», текстового вируса и экстенсивного использования языка, с помощью которых мы пересмотрим устоявшуюся оппозицию исполняемого и неисполняемого. В дополнение к существующим понятиям мы рассмотрим дополнительные термины выполнения (performance) и исполнения (execution), которые помогут пересмотреть практику mez в современном медиатехническом контексте, отличном от существовавшего во время создания этих текстов.

# **Глава 1. Концептуализация современных технологий в теории медиа и электронной литературе**

Исследование электронной литературы предполагает не только знание различных методов интерпретации текстов, но и пристального внимания к их материальным условиям производства, хранения и репрезентации. Медиатехническая среда, в которой существуют эти тексты, влияет на их рецепцию не меньше, чем их формальные характеристики. Для того, чтобы рассмотреть тексты mez с точки зрения медиатеории, следует дать характеристику основным существующим способам концептуализации современного медиатехнического окружения, рассмотреть ключевые понятия, связанные с компьютерными науками и то, как они интерпретируются медиатеоретиками, дать определение понятию электронная литература и поджанру «codework», в котором созданы тексты изучаемого автора.

**1.1 Теории медиа**

Существует два подхода к изучению медиа: первый из них подразумевает эмпирическое изучение средств массовых коммуникаций, их социального, политического и культурного значения. Второй, интерпретативный, занимается исследованием технического, опосредованного существования культурных артефактов, а также влияния их материальной основы на производство значения и их рецепцию сквозь призму теории литературы, философии техники, культурных исследований и других методологий гуманитарного знания. В нашем исследовании мы будем придерживаться второго подхода. Поскольку теория медиа включает в себя множество различных методологий, мы рассмотрим общие для них положения, и конкретизируем различные способы концептуализации современного медиатехнического окружения в целом, представленные в работах нескольких авторов.

В первом приближении, все теоретики медиа сходятся на существовании фундаментальной для человека способности опосредовать знание и восприятие с помощью различных инструментов, передавать его на расстояние и хранить на различных материальных носителях, а также интериоризировать элементы медиаокружения в свои когнитивные структуры. Одним из первых ученых, сформулировавших такую точку зрения, был канадский культуролог и философ Маршалл Маклюэн. В книге «Понимание медиа», он описывает основные свойства современной ему медиа-технической ситуации, такие как «вложенность» одного средства коммуникации в другой, когда содержанием одного медиума (например, кино) является другой медиум (роман или пьеса), важную роль средства передачи информации, влияющую на восприятие передаваемого содержания (описываемая известным афоризмом «медиа это сообщение»), и проекцию человеческих способностей вовне, расширение чувств, которые одновременно и лишают человека той способности, которую он экстернализировал. Также он подчеркивает информационный характер этих вынесенных вовне способностей: «Помещая с помощью электрических средств коммуникации свои физические тела в свои вынесенные наружу нервные системы, мы приводим в действие динамический процесс, в ходе которого все прежние технологии, являющиеся просто-напросто расширениями рук, ступней, зубов и механизмов поддержания температуры тела — все эти расширения наших тел, включая города, — будут переведены в информационные системы». [Маклюэн 2003: 69]. Подводя итоги, можно сказать, что Маклюэн своими работами заложил фундамент для дальнейших исследователей медиа, разрабатывающих свои собственные концепции не только на основе работ Маклюэна, других ранних теоретиках медиа (таких, как В. Беньямин, Г. Иннис и др.) и текстах философов о технике, но и в соответствии с появлением и развитием новых медиакоммуникационых технологий. Далее мы рассмотрим основные положения таких известных медиатеоретиков, как Кэтрин Хэйлс, Бернар Стиглер, Фридрих Киттлер, Лев Манович и Марк Хансен. Их тексты в целом описывают поле возможных подходов к проблеме медиа, необходимых для исследования в рамках данной работы.

Кэтрин Хэйлс с помощью понятия «постчеловеческого» (posthuman) описывает размывание границ между человеческим и машинным, кибернетическими механизмами и биологическими организмами. Согласно ее исследованиям, живые и неживые акторы собираются в когнитивные ассамбляжи, находящиеся в постоянном изменении сети, позволяющей циркулировать информации, интерпретациям и значениям между человеческими и нечеловеческими акторами. Термин когниции она понимает в широком смысле, не как отдельную способность, а как спектр различных возможностей, «процесс интерпретации информации в контекстах, соединяющих ее со значениями». [Хэйлс URL]. Когницией обладают практически все существа на планете, но в разной степени сложности. Например, растения обладают минимальными когнитивными способностями, а млекопитающие и люди гораздо более сложными когнитивными системами.

По мнению философа и медиаисследователя Бернара Стиглера, медиа (в единственном числе) – это «онтологическое свойство становления человеком (humanization) – конститутивная операция экстернализации и изобретения». [Митчелл, Хансен 2010: 8]. Медиа во множественном числе, или медиумы, это конкретные способы экстернализации той или иной способности человека в его взаимодействии с окружающим миром. Такого рода экстернализации не просто дают возможность по-разному работать с действительностью, но и изменяют тот материал, к которому применяются. Наш понятийный аппарат также подвергается изменениям, акцент с одних понятий переносится на другие. Можно сделать вывод, что медиатеория занимается исследованием не только отдельных медиумов, их свойств и влияния на информацию, но и более общей категории медиа: «<…> исследования медиа занимается как субъективной, ментальной жизнью, так и машинами, кодами и сообществами. Она рассматривает не только расширения человеческого сенсориума, но также их интроекцию в структуры чувствования и формы жизни, конституирующие человеческую субъективность и общность». [Там же: 11]. Человек эволюционировал не только биологически, но и культурно, экстериоризируя знания и навыки в виде технических инструментов и культурных артефактов. Используя идеи своего учителя Жака Деррида, он пишет, что процесс восполнения всегда связан с конкретными технологиями хранения и передачи.

Для другого медиаисследователя, Фридриха Киттлера, медиа «определяют нашу ситуацию», т.е. полностью детерминируют взаимодействие человека с окружающей средой. В своей книге «Оптические медиа» он утверждает, что у технических медиа существует всего три функции: накопление, обработка, и перенос информации. В главном на сегодняшний день медиуме, персональном компьютере, все эти три функции сочетаются, причем не только на высоком уровне абстракции (интерфейс и языки программирования высокого уровня), но и на самом низком, на уровне машинного кода: «Все программы перед исполнением должны быть превращены в последовательность таких команд, которые обычно не сложнее, чем, например: сложить два числа; проверить, не является ли число нулем; скопировать блок данных из одной части памяти компьютера в другую. Эти примитивные команды в совокупности составляют язык, на котором люди могут общаться с компьютером. Такой язык называется машинным». [Таненбаум : 20]. Для Киттлера сущностью медиа и является машинный код (о чем говорит его известная статья «Программного обеспечения не существует»), не предназначенный для восприятия человеком и не адаптируемый под него.

Очень точно место компьютера в современной ситуации выразил Лев Манович в своей известной книге «Язык новых медиа»: «компьютер воплощает в себе функции шоу-рума и завода одновременно: одна и та же машина и создает, и показывает медиа. С другой стороны, подобная скорость достигается за счет того, что современные медиа существуют не в виде материального объекта, а в виде формата данных, которые можно передать со скоростью света». [Манович 2018: 71]. В своей книге «Язык новых медиа» он приводит пять принципов новых медиа: цифровая репрезентация (вся информация представлена в числовом, дискретном формате, и подвергается алгоритмическим манипуляциям), модульность (если элемент становится частью большего элемента, его внутренняя структура не меняется, он доступен в любой момент независимо от включения в б*о*льшую сборку), автоматизация (многие процессы можно выполнять без непосредственного вмешательства), вариативность (объект никогда не завершен, может существовать во множестве версий) и транскодинг (перевод из одного формата в другой, часто из аналогово в цифровой, при этом цифровой файл имеет два «слоя»: культурный и компьютерный). Все медиатехнологии он делит на две ветви: те, которые развивались для усовершенствования репрезентации, и те, которые развивались для улучшения коммуникации. В персональном компьютере эти два вектора сходятся, что приводит к пониманию компьютера как метамедиума, на котором контент создается и отображается. Также меняется роль реципиента – из пассивного наблюдателя он становится активным пользователем, объекты становятся интерфейсами. На экране, на котором происходят интеракции, поочередно сменяются два режима: контроля и репрезентации. Первый из них подразумевает определенные действия, которые надо совершать (например, печатать текста, переносить файл из одной папки в другую). Во втором режиме экран дает иллюзию трехмерного пространства (например, при просмотре фильма). Кроме изучения визуального аспекта новых медиа, Манович подробно останавливается на компьютерном слое, логике баз данных и алгоритмов. Все программное обеспечение можно разделить на неструктурированные базы данных и алгоритмы, работающие с ними. Для них Манович находит аналогии из лингвистики и сравнивает их с парадигмами и синтагмой. В его интерпретации парадигма занимает ведущую позицию, материализуется, поскольку синтагм (алгоритмов) может быть несколько, и они дематериализованы.

Теория медиа Марка Хансена отличается от уже обозначенных подходов к медиа. Ключевое место в его теории занимает изучение того, как телесный опыт проявляет себя через взаимодействие с цифровым окружением, а конкретно с цифровыми образами. Поскольку медиа теперь не занимаются фиксацией реальности, а становятся различными интерфейсами для отображения неструктурированных данных в виде изображений, текста или видео, теряют свою материальную специфичность, в этой ситуации тело становится пространством отбора различных цифровых образов, основой для их развертывания. Единицами такого восприятия можно назвать цифровые образы. Само это понятие должно быть расширено до всего спектра воспринимаемой информации, не только визуальной, но и телесной, проприоцептивной и гаптической. Согласно теории Хансена, цифровые технологии не дематериализуют уже существующий опыт, но актуализируются в телесном восприятии субъекта, не ограниченным только визуальной или языковой информацией.

Рассмотрев основные способы изучения медиа, следует перейти к понятиям, в которых отражается проблематика современной медиатехнической среды. С их помощью можно выделить аспекты, которые конструируют возможность производства и рецепции электронной литературы. Эти термины можно разделить на три группы по основным операциям, производимым над информацией, как это делает Ф. Киттлер в своей книге «Оптические медиа». К этим операциям относятся обработка, хранение и передача. Вначале рассмотрим основные понятия информации и шума, затем перейдем к терминам, относящимся к каждой отдельной группе.

**1.1.2. Информация и шум**

Можно выделить два подхода к определению информации: количественный и качественный. Количественное определение информации дал математик Клод Шеннон в статье «Математическая теория связи». По Шеннону, информация – это вероятность выбора сообщения из ограниченного набора вариантов сообщения: чем меньше их количество, тем меньше возможных выборов и тем меньше информации эти выборы несут. Например, выбор между «да» и «нет» дает один бит информации, поскольку возможность выбора ограничена двумя вариантами. Чем больше у субъекта вариантов выбора, тем большее количество информации он получает. Шеннон пишет, что сообщение несет определенное значение, но для инженера при проектировании средств коммуникации важно сохранить целостность передаваемого сообщения, свести к минимуму воздействие шума. То, что передается, не учитывается. Соавтор Шеннона, Уивер, дополняет его теорию семантическим компонентом, но подчиняет его техническим свойствам сообщения.

Другой подход к определению информации, в котором значение сообщения выдвигается на первый план, можно назвать качественным. Например, Грегори Бейтсон определяют информацию как «различие с определенными последствиями». [Хуэй 2020: 180]. Для французского философа техники Жильбера Симондона информация означает «различие, которое производит определенный эффект в индивиде (рассматриваемом как система), способный привести к процессу индивидуации или, по крайней мере, ему способствовать». [Там же: 181].

Дональд Маккей выделяет два аспекта процесса коммуникации: выбор и конструирование. Первый из них можно свести к определению информации Шеннона, второй из них независим от технической конфигурации и обусловливается контекстом поведения получающего информацию, того как он ее структурирует. В теории информации Маккея паттерны не присущи самой информации, как в теории Шеннона, но задаются воспринимающим.

Определение современного медиатеоретика Брюса Кларка схоже с тем, которое дает Маккей: «Информация – это виртуальная структура, зависимая от распределенных режимов кодирования/декодирования, в которых она может функционировать». [Кларк 2010: 155]. Она имеет релятивистский характер, поскольку «количественное исчисление информации различается в зависимости от позиции наблюдателя». [Там же: 155]. Информации не существует в природе, пока не появится субъект, который сможет сконструировать ее и выделить среди шума среды, в которой он/а находится. Кларк пишет, что информация не является просто передаваемым от отправителя к получателю данными, но результатом самореферентных процессов в когнитивных системах, вне их она не существует.

С понятием информации связано понятие шума, того, что приходит вместе с сообщением, при этом не являясь изначально ее частью. Обычно шуму противостоят с помощью повтора ключевых частей сообщения, жертвуя при этом производительностью, но взамен получая гарантию, что сообщение дойдет без потерь. Нужно отметить, что шум — это не отсутствие информации или искажение, а ее переизбыток, те неожиданные элементы полученного сообщения, которые отсутствовали в изначальном сообщении. В классической теории информации понятие шума имеет второстепенное значение, но если сфокусироваться не на проблемах передачи, а записи и сохранения, присущим для современных медиа, то шум приобретает значение, поскольку избыточность информации может иметь значимость, например, в некоторых видах искусства: в шумовой музыке, абстрактном экспрессионизме и в др: «Включая большую неопределенность в сообщение, шум можно рассматривать не просто как потерю ясности или производительности, но как форму информации о медиаокружении, увеличение коммуникативной неопределенности с потенциалом к введению иной значимой информации в процесс коммуникации». [Там же: 164]

Переходя из математики в кибернетику и другие области знания, понятие информации дополнялось другими понятиями, важнейшим из которых является обратная связь. Это происходит из-за переориентации роли информации в системах – от хранения и передачи к использованию и циркуляции информации внутри системы. Обратную связь (фидбек) можно определить, как цикл циркуляции информации, когда данные на выходе снова передаются в систему как входные данные. Обратная связь бывает отрицательной и положительной. При отрицательной обратной связи система стабилизируется, новая информация не привносится в систему. Положительная обратная связь характеризуется тем, что при возвращении данных в систему добавляется новая информация.

# **1.1.3 Исполнение**

Исполнение – очень важная категория для исследований медиа и конкретно исследований программного обеспечения (software studies). Оно тесно связано с понятием кода, с возможностью записи команд на языке программирования и выполнение этих команд компьютером. Исполнение определяют как «автоматизированный акт периодического последовательного повторения набора читаемых машиной инструкций, в рамках которых любые действия будут исполняемыми (интерпретируемыми бинарным дешифратором (logical decoder), чьи сигналы счетчик команд (instruction pointer) адресует тому компоненту, над которым проводится операция) согласно параметрам активного [в данный момент] набора команд и повторение этих циклов до тех пор, пока они исполняемы или до того момента, когда больше не будет инструкций для исполнения (счетчик команд указывает на конец последовательности инструкций)». [Снодграсс 2018: 239-240].

Хотя это определение относится к формальным языкам программирования, возможность слова выполнять действие концептуализировалась разными учеными, философами и литераторами задолго до появления компьютера. Флориан Крамер в своей книге «Слова стали плотью» описывает такого рода попытки заставить слово «воплотиться»: магические практики, лурианская каббала, протеическая и комбинаторная поэзия, и т.д. Александер Гэллоуэй в книге «Протокол: существование контроля после децентрализации» называет программный код «единственным исполняемым языком, т.е. первый язык, влияющий на материю» [Гэллоуэй 2004: 244]. Код – это инструкция, предназначенная только для машины. Для него не существует различия между исходным кодом и исполнением. Другой медиатеоретик, Вэнди Чан, не соглашается с Гэллоуэем и усложняет схему превращения текста в действие. Она пишет, что исходный код становится таковым только постфактум, когда он выполнится с помощью других программ (т.е. другого кода), и в этом состоянии их трудно отличить, поскольку код пересобирается по ходу исполнения. Код не разворачивается линейно, а проходит через промежуточные результаты, возвращается к уже пройденным этапам и может самоизменятся без участия человека. Таким же образом исполнение описывает китайский философ Юк Хуэй, когда противопоставляет линейное, последовательное исполнение программы, характерное на ранней истории компьютерных вычислений, и современное рекурсивное программирование: «Концепт фидбека, введенный в кибернетике, задал новую темпоральную нелинейную структуру, похожую на спираль. В этой структуре путь к конечной цели будет не линейным, но постоянным саморегулирующимся процессом». [Хуэй 2018: 31]. Код является скорее «ресурсом» (re-source), к которому обращаются по мере надобности, а не исходной точкой (source).

Другая важная тема, которую затрагивает Чан в своей книге – это тема абстракции, т.е. создания максимально эффективных и общих способов работы с кодом и данными. Она приводит высказывания известного ученого Эдсгера Дейкстры о необходимости «уменьшения концептуального разрыва между статичной программой и динамическим процессом [ее исполнения], сделать сходство программы (разворачивающейся в текстовом пространстве) и процесса (развертывающегося во времени) как можно более очевидным». [Чан 2011: 36]. Программист не должен вдаваться в детали того, как работает та или иная часть кода или системная библиотека, которые он использует, видоизменяет или поддерживает в рабочем состоянии. Важным становится умение манипулировать готовыми ресурсами. В таком понимании программирования стирается разница между исходным кодом и его исполнением. Эту ситуацию и критикует Вэнди Чан.

Далее ее критику развивает Дэвид Готье в статье «О командах и исполнении: тираны, призраки, бродяги». Он рассматривает понятие команды как таковой и его основания в том или ином символическом регистре: логическом, арифметическом, машинном и в др., и каким образом можно помыслить исполнение, не смешивая его с символической записью кода. Для этого он вводит понятие «герменевтического цикла» (hermeneutic loop), состоящей из интерпретации, действия и интерпретации совершенного действия. Во время исполнения инструкция задает параметры действию и после его исполнения интерпретирует его соответствие или несоответствие исходным параметрам, чтобы в зависимости от интерпретации продолжить исполнение. В этом герменевтическом цикле простое действие одновременно претерпевает воздействие и скрывается интерпретацией. В этом стирании действия, заменой его интерпретацией Готье видит проблему, поскольку информатика (computer science) находится в метапозиции по отношению к исполнению – она учитывает только входящие и исходящие данные, но не сам процесс исполнения.

В контексте литературы и современного искусства понятие исполнения рассматривается в широком смысле: исполнение программы происходит не только на компьютере, но и в уме у пользователя. Любой алгоритм производства текста (как, например, известное произведение Т. Тцары) также является программным обеспечением: «Если процесс, разработанный Тцара, будет переведен на язык Perl или С с оригинального языка - французского, он не будет просто переводом «чего-то» на язык программного обеспечения, но – транскрипцией немашинного программного обеспечения в машинное». [Крамер URL]. Другое важное замечание Крамер делает в статье «Поэзия программного кода» (Program Code Poetry), где описывает склонность теоретиков и практиков цифровой поэзии уделять больше внимания интерфейсу, а не коду. Такое рассмотрение практик code poetry учитывает их визуальные составляющие, но не «художественные преимущества потенциального машинного исполнения любого цифрового кода и как минимум двух прочтений этого кода в качестве исходного кода и результата его исполнения (output)». [Крамер URL]. К таким же выводам приходит медиасследователь Дэннис Тэннон, когда пишет про цифровой текст: «Я пойду дальше остальных в мнении о том, что цифровой текст *это* код, в том смысле, что он постоянно парсится и потенциально исполняем». [Тэннон 2017: 13].

Интересный подход к оппозиции исполняемое/неисполняемое предложил австралийский художник и медиаисследователь Питер Уайлдмен. В своей работе «Деконструируя рабочий код» он обращается к полемике Джона Кэйли с Ритой Райли и Кэтрин Хэйлс насчет понятия исполнения и на примере языка программирования «Processing» показывает, как будто бы неисполняемые фрагменты текста, введенные в текстовый редактор интегрированной среды разработки, при запуске процесса компиляции помечается как исключение, мешающее дальнейшей работе программы. Затем создается объект (основной элемент объектно-ориентированных языков программирования), называемый «объект исключения» (exception object), затем он перемещается в среду выполнения и появляется на экране в виде сообщения об ошибке. Таким образом, неисполняемый (в традиционном определении этого слова) фрагмент текста становится исполнением процесса поиска и нахождения ошибки, хотя и на минимальном уровне. Вместо жесткой границы между исполняемым и неисполняемым Уайлдмен характеризует отношение взаимодополнения этих двух понятий, поскольку наборы символов вписываются в структуру кода, но не на уровне содержания (как это делает Кэйли, когда смотрит только на результат исполнения) а на уровне процедуры – когда создается новая функция, вычисляющая нерабочий код, сигналящая о его присутствии и приостанавливающая работу программы. Насчет текстов mez он пишет, что в будущем они могут стать исполняемым кодом для другого языка программирования, поскольку он имитирует структуру программного кода на уровне синтаксиса, к тому же, ее произведения подходят под понятие «псевдокода». В «прямом программировании» каждая программа была предназначена только для одной машины. «Псевдокодом» назывался текстовая запись того, что делает та или иная программа (которая вводилась и исполнялась посредством ручного переключения кабелей в машине). Когда появилось структурное программирование, и программное обеспечение стало возможным отделять от аппаратного обеспечения, то «псевдокод» стал исходным кодом – высокоуровневой символической записью компьютерных команд, не привязанных к одной машине, открытой для дальнейшего изменения и повторного использования.

В целом, можно сделать вывод, что медиаисследователи расширяют само понятие и пытаются рассматривать исполнение вместе с другими контекстами, на которые работа исполнения оказывает влияние: «Исполнить не значит просто запустить скрипт для осуществления определенного действия. Это также запустить множество материально-политических отношений и преобразований. В то же время — это скорее сплав взаимодействующих событий и случайностей, разыгрываемых в перформансе исполнения». [Габри 2018: 297].

# **1.1.4. Программный код**

Понятие кода является одним из ключевых понятий в информатике (computer science). Оно не только охватывает множество объектов и операций, предполагающих кодирование, но и задает определенную рациональность, отличную от предшествующей инструменталистской рациональности. Поэтому следует вначале описать то, как информатика конструирует способы восприятия и изменения мира, а затем рассмотреть понятие кода как основу подобного конструирования.

Медиатеоретик Дэвид Бэрри так описывает новый тип рациональности, присущий информатике: «вычислительная рациональность — это форма мышления, которая осуществляется при помощи других нечеловеческих агентов, но эти объекты сами способны проявлять агентность, выполняя вычисления, принимая решения или оказывая коммуникативную поддержку пользователю». [Бэрри 2011: 13]. Компьютерные вычисления ограничиваются наборами символьных дискретных данных и определенными алгоритмами, но уже закодированное имеет неограниченный потенциал к пересборке, трансформации и фильтрации.

Рассмотрим теперь понятие кода и как оно концептуализируется в информатике и в литературе. Любая программа является «последовательностью инструкций, закодированных в виде цифровых последовательностей» [Брукшир, Бриллов … 434]. Такой способ кодировки называется машинным кодом. Но поскольку человеку сложно работать с большими массивами двоичных чисел, был изобретен язык ассемблера – язык, в котором команды записывались не в шестнадцатеричной системе счисления, но в виде мнемонических команд (например, в таком виде: MOV R5, R8). Несмотря на преимущества языка ассемблера, он был машинно-зависимым, т.е. программы, записанные на одной машине, нельзя было запускать на другой, поскольку мнемонические команды в ней могли отображаться другими буквенными значениями. Поэтому позднее были разработаны языки программирования высокого уровня, где абстракции подверглись уже более объемные фрагменты кода. Языки программирования высокого уровня являются лишь репрезентацией тех же процессов, закодированных в машинном коде. Написанная программа исполняется после перевода в машинный код: или в процессе компиляции, когда создается перевод программы в машинный код, а затем ее можно исполнять, или в процессе интерпретации, когда программа исполняется строчка за строчкой. Кратко можно подытожить что история языков программирования развивалась от работы с компьютером, при которой программисту требовалось понимать все особенности работы машины, до того момента, когда машины должны учитывать все особенности человеческого восприятия и мышления.

От истории следует перейти к основным концепциям, существующие в большинстве языков программирования высокого уровня. Они бывают разной степени абстракции, «начиная от простейших (констант, литералов и переменных) через умеренно сложные (операторы, выражения и управляющие структуры) до понятных лишь посвященным (процедурные единицы, модули и библиотеки)». [Там же: 448]. Программа состоит из операторов трех видов: операторов объявления, выполняемых операторов и комментариев. Первые из них описывают данные, которые в дальнейшем будет использовать программа. Вторые описывают этапы использования алгоритма (например, операторы присваивания присваивающие переменной значение, управляющие операторы, изменяющие порядок выполнения программы, и др.), а в комментарии на естественном языке дается пояснение тому или иному аспекту программы.

Другие важные концепции языков программирования ­­– переменные, типы данных и структуры данных. Переменные – это адресуемая область памяти. Тип данных определяет то, как их нужно интерпретировать и какие операции могут проводиться с данными. Структуры данных определяют общую форму их организации.

Одной из самых важных концепций языков программирования является функция – «набор инструкций, необходимых для решения определенной задачи, который может использоваться как абстрактный инструмент другими программными единицами». [Там же: 468]. Практически любая программа содержит в себе несколько функций, совершающих определенные задачи.

Теперь необходимо перейти к концептуализации языков программирования, существующих в исследованиях софтвера. В книге «Философия софта – код и медиация в цифровую эпоху» Дэвид Бэрри дает определение кода в контексте компьютерных наук: «код можно понимать как механизм, управляющий и изменяющий символьные данные с помощью рекомбинации, арифметических операций или перемещении данных из одного места в другое <…> код операционален и производит результат (иногда в результате выполнения промежуточных целей), часто в процессе повторения и условных операторов». [Бэрри 2011: 52].

Бэрри, как и другие исследователи, указывает на важность семантической составляющей при чтении и написании кода. Хотя изначально кажется, что первоначальное значение код обретает только в исполнении машиной, для пишущего код все равно существует некоторая свобода выражения в различных способах достижения одной цели (обфускация) и визуальной составляющей кода (perl poetry). В творческом аспекте это подчеркивается акцентированием некоторых программистов на «самой программ[е], а не результате ее работ[ы]». [Орехов URL].

Другая важная тема, которую исследует Бэрри в книге – это темпоральность кода. Темпоральность выражается в последовательном выполнении огромного числа команд за очень короткий промежуток времени, не воспринимаемый пользователем. Время исполнения кода идет гораздо быстрее, чем время человеческой жизни, и синхронизировать их призван интерфейс, «<…> важная часть проектирования компьютерных систем, скрывающая от пользователя процесс исполнения кода <…>». [Там же: 98].

Важным понятием, связанным с программным кодом, является понятие алгоритма. В сборнике «Software Studies: A Lexicon» под редакцией Мэтью Фуллера в одноименной статье Эндрю Джоффи, ссылаясь на ученых из области «computer science», описывает алгоритм как описание метода, выполняющего некое задание. Важно подчеркнуть, что алгоритм – это абстракция, существующая независимо от применения в той или иной аппаратной среде. Теоретически, алгоритм может быть встроен в литературное произведение или создавать его (см. эксперименты группы УЛИПО).

В отношении к культуре у кода двойственная позиция – с одной стороны, код выполняет важную функцию коммуникации человека с машиной, с его помощью создается и поддерживается инфраструктура современного медиатехнического окружения, в основе которой не только формальные языки программирования, но и множество культурных, визуальных кодов, адаптирующих пользователя к работе в медиатехническом окружении.

После рассмотрения необходимых понятий, связанных с обработкой информации, перейдем к выделению необходимых для исследования терминов, описывающих её хранение.

# **1.1.5. Знак**

Понятие знака является центральным для многих областей знания, связанных с естественными и функциональными языками. Для начала, приведем определение знака в семиотике как наиболее общее для гуманитарных наук: «Знак (лат. signum) — это материальный, чувственно воспринимаемый предмет (вещь, явление, действие, признак), выступающий

в качестве представителя (заместителя, репрезентанта) другого предмета, свойства или отношения и используемый для получения, хранения, переработки и передачи информации.». [Мечковская : 23]. Если применять это определение в контексте современных медиатехнологий, то нельзя не учитывать важных аспектов, которые привносятся новыми технологиями в понимание знака.

Понятие знака в контексте медиа теории следует рассматривать вместе с медиатехническим окружением, в котором он существует. Многие медиатеоретики и исследователи электронной литературы обращаются к теме знака, чтобы уточнить его изменчивую функцию в текстах на экране. Одной из первых эту проблематику затрагивает Кэтрин Хэйлс. В статье «Печать плоская, код глубок» она описывает то, каким способом означающее репрезентируется на экране: «В компьютере означающее существует не как стабильный знак, нанесенный на плоскую поверхность, но как изображение на экране, продуцируемое несколькими уровнями кода, соотносящимися по строгим правилам, от электронных полярностей, коррелирующих с потоком двоичных сигналов до битов, коррелирующих с двоичными числами, переводимыми в цифры, коррелирующими с командами высокого уровня и т.д. Даже когда электронный гипертекст изображает отдельные знаки, на самом деле они являются временными изображениями, <…> постоянно обновляемыми, создающими иллюзию неподвижного изображения». [Хэйлс URL].

Брайан Ким Стефанс противопоставляет идею «игры означающих», с помощью которой описывается функционирование знака в современной литературе, его пролиферацию и независимость от субъекта письма или чтения, отсутствие одного установленного значения и «кибернетический обмен», который указывает на различные уровни репрезентации «письменного» знака на экране и подчеркивает совместную работу автора и той программной среды, в которой создается или репрезентируется текст. Также он делает важное замечание насчет несоответствия феноменологического опыта играющего в компьютерную игру и процессов, конструирующих этот визуальный опыт: «Геймер верит в то, что разворачивающееся на экране [пространство] *синтагматично,* словно единичное высказывание, но на самом деле оно *парадигматично:* пустые структуры, заполняющиеся и опусташающиеся в ходе алгоритмических процессов». [Стефанс URL]. Это различие применимо не только в случае компьютерных игр, но и в целом для опыта взаимодействия с компьютером, поэтому важно его учитывать для изучения литературного текста на экране.

Кроме пересмотра уже существующего понятия знака существуют попытки дать новое определение гибридному характеру знака в современной медиареальности. Дэннис Тэннон в своей книге «Простой текст: поэтика вычислений» приводит несколько интересных понятий, таких как «поэтика вычислений», которая занимается интерпретацией мельчайших текстовых компонентов, а также понятие «программатического» знака, которое следует рассмотреть подробней, поскольку оно развивает уже обозначенные подходы к понятию знака.

Программатический знак имеет двойственную природу: экранную репрезентацию и место в памяти компьютера. На экране он подражает известным нам знакам, предоставляет инструменты для манипулирования ими. На уровне памяти визуальная репрезентация является серией структурных трансформаций: «механическое нажатие клавиши преобразуется в (2) электрический сигнал, который (3) оставляет электромагнитные следы в памяти компьютера, (4) превращающиеся в фазы жидких кристаллов на экране, оставляющие за собой (5) знаки, отображающиеся на экране в виде света». [Тэннон 2017: 24]. То, что на жестком диске хранится в разделенном виде, с помощью языков программирования собирается на уровне репрезентации в целое. Эти репрезентации являются скорее метафорами внутренних процессов, скрытых от человека. Программный код контролирует этот процесс перевода, а значит, знак на экране является лишь фрагментом сложной системы контроля и коммуникаций между различными компьютерными подсистемами – он отсылает не только сам к себе, другим письменным знакам или к объекту внешнего мира, но к другим скрытым от пользователя математически формализованным кодам, и уже известным в культуре способам работы с символическим.

Подводя итоги, можно сказать, что при рассмотрении знака в медиатехническом окружении нужно учитывать сложные отношения между видимой поверхностью интерфейса и скрытыми кодами, конструирующими репрезентацию и контролирующими её непрерывность для пользователя.

**1.1.6. Память**

Память – это одна из ключевых категорий не только в гуманитарных науках, но и в информатике и медиатеории. У этого понятия очень долгая история в философии, от Платона и Аристотеля до Фрейда и Деррида. В медиатеории существует несколько подходов к проблеме памяти. Вначале мы рассмотрим ее во многом внешний характер, то, как она эксериоризируется с помощью технических приспособлений, а затем рассмотрим аналогию между человеческой памятью и памятью компьютера, которая во многом задала культурные и технические способы концептуализации памяти в современном медиатехническом окружении.

Французский философ Бернар Стиглер описывает историю понятия памяти как взаимодействие овнешнения памяти с внутренним процессом запоминания. Он проводит различие между техниками запоминания, присущими индустриальной эпохе (фотография, фонография, кинематограф) и техниками постиндустриальной эпохи, связанными с вычислительной техникой (компьютеры, смартфоны). Если технологии индустриальной эпохи разделяли производство и потребление, то современные технологии убирают границу между потребителем и производителем: субъект принимает процедурные правила медиатехнологического окружения, согласно которым, процесс принятия и создания сообщения симметричен. Исторический процесс перехода от индустриальных технологий памяти к постиндустриальным Стиглер описывает как переход от мнемотехник к мнемотехнологиям: «Это особенно очевидно в переходе от мнемотехник к мнемотехнологиям – от индивидуальной экстериоризации функций памяти к крупномасштабным технологическим системам или сетям, организующим воспоминания». [Стиглер 2010: 64]. По Стиглеру, память стала одним из ключевых элементов постиндустриального производства. Огромное количество разнообразных медиатехнологий ведет скорее не к потере возможности помнить, но к дислокации памяти, размещении ее в технологических системах, слишком сложных для отдельного пользователя и не дающих ему полный контроль над тем, что остается в памяти, в каком виде она существует и кто ей еще может воспользоваться. Процесс, благодаря которому появление таких технологий стало возможным, Стиглер называет грамматизацией: деление раннее непрерывных процессов на дискретные элементы. Примером может быть появление письменности, делящей поток речи на дискретные элементы – фонемы. Прошлые и настоящие формы субъективации конфигурируются благодаря специфичному для каждой эпохи процессу грамматизации. Постепенно этот процесс переходил от языка на другие сферы человеческой деятельности: когнитивные, телесные и т.д. Процесс запоминания не нейтрален: он проходит через селекцию нужной информации и забывание ненужной, содержание в памяти выстраивается в иерархии согласно важности и доступу к ним. В современную эпоху этот процесс осложняется: «Однако, нечто абсолютно новое происходит когда условия запоминания, а именно, критерии стирания, отбора, забывания, ожидания, ретенции-протенции – одним словом, темпорализации – концентрируются в техно-индустриальном комплексе <…>». [Там же: 77]. Память становится ресурсом, который может быть использованным не только самим субъектом, но и тем, кто контролирует хранение информации в современных медиасредах. Такое функционирование памяти предзадает поведение субъекта, его знание о способах работы с информацией и ожидание того, что его действия будут фиксироваться даже без его ведома.

От более общего описания понятия памяти в современном медиатехническом окружении перейдем к метафоре человеческой памяти как памяти компьютера, которая имеет некоторые практические последствия при разработке современного программного обеспечения.

Ученые-когнитивисты считают, что лучшей моделью человеческой памяти является память компьютера. Хотя такие представления уже давно оспорены более поздними исследованиями, эта упрощенная модель используется в информатике, но немного в другом виде. Для компьютерных наук память в первую очередь связана с управлением и распределением дискретных единиц, таких как списки, числа, таблицы. Более сложные механизмы памяти, которые описаны в некоторых литературных текстах (например, роман в семи частях М. Пруста «В поисках утраченного времени») не затрагиваются информатикой. Компьютерные науки создают специфическую модель человеческой памяти, занятой расчетами, контролем и организацией: «<…> все компьютерные вычисления глубоко укоренены в метафорах и прагматических установках, присущих бюрократии. <…> Когда истоки этих установок забываются, теряется взгляд на специфичные образы памяти и мышления, материализованные в дизайне софтвера и принципах его проектирования». [Сэк 2008: 190].

Таким образом, память является не столько внутренней способностью субъекта накапливать нужную ему информацию, но сложную конфигурацию и взаимовлияние вынесенных вовне методов сохранения и передачи информации, и когнитивных механизмов субъекта.

**1.1.7. Коммуникация**

От понятий, связанных с обработкой и хранением информации перейдем к способам и условиям ее передачи. Начать следует с понятия коммуникации, охарактеризовать его не в самом общем смысле как процесс передачи сообщения от отправителя к получателю, но с опорой на существующие способы концептуализации этого понятия в информатике и медиатеории, а также в многочисленных видах передачи сообщения и медиатехнические условия такой передачи.

В контексте теории медиа нужно учитывать две важные коннотации этого термина:

1. «общность (the communality) коммуникации — ее сложная множественность и обусловленность в большей степени социальными факторами, а не индивидуальными намерениями.
2. материальность коммуникации — материальные и технические инфраструктуры, необходимые для передачи сообщений или любой другой информации». [Кларк 2010: 129]. Общность коммуникации следует учитывать, поскольку ее системы слишком сложны для полного понимания субъектом и требуют более подробного рассмотрения, включающего не только индивидуальный опыт, но и социальные и культурные факторы. Материальность коммуникации важны для исследования поскольку формируют то, как будет передаваться сообщение и какие технические и материальные аспекты влияют на процесс коммуникации.

Роберт Крейг, автор словарной статьи о коммуникации в «Энциклопедии риторики», описывает семь традиций изучения коммуникации: риторическую, семиотическую, феноменологическую, кибернетическую, социопсихологическую, социокультурную и критическую, и два способа концептуализации этого понятия: модель передачи, которая подразумевает коммуникацию как процесс передачи сообщения в символической форме, и конститутивная модель, которая рассматривает элементы коммуникации не как предписанные заранее, но постоянно модифицируемые в процессе коммуникации. В исследовании будет использоваться вторая модель, поскольку она лучше описывает современную ситуацию, в которой существуют процессы коммуникации.

В конститутивной модели характер современных медиакоммуникаций отличается тем, что информация не только передается, но сохраняется с последующей возможной обработкой. Содержание сообщения или информация – это не неизменяемая сущность, которую можно точно измерить и посчитать, но относительная величина, поскольку на нее влияют то, как отправители и получатели конструируют сообщение и как отделяют нужную информацию от шума.

Также пересматривается роль коммуникантов в процессе передачи и получения сообщений через критику понятия интерсубъективности. Это понятие ввел немецкий философ Юрген Хабермас. Согласно Хабермасу, с помощью универсального лингвистического кода коммуниканты могут переводить содержание сознания и передавать его без всяких потерь. Критикуя это понятие, Брюс Кларк в своей статье о коммуникации опирается на теорию систем немецкого социолога Никласа Лумана: «С этой точки зрения, концепт интерсубъективности некогерентен, потому что стирает границу между операциональными замкнутостями двух разных типов систем: социальных и психологических. Базовая посылка модели Лумана – «операциональная закрытость», репрезентирующая аутопоэтическую систему самореференциальной. Такие системы не могут действовать за пределами своих границ, поскольку закрытость является условием их возможности. Разум <…> тоже аутопоэтическая система, обрабатывающая восприятие и (воз)производящая сознание. По определению, психика также операционально закрыта, и сознание не может напрямую влиять на свое социальное окружение. В то же время, сообщения, (ре)продуцируемые социальными системами не могут напрямую повлиять на сознания, находящиеся в этой социальной среде». [Там же: 138]. По Луману, информация не передается, а конструируется и реконструируется по ходу коммуникации. Язык в этой схеме не способ передачи, работающий поверх других систем, но одна из структур медиации сообщения, включенная в другие сети и протоколы.

Подведем итоги. Коммуникация является не закрытым однонаправленным процессом, происходящим между коммуникантами, но петлей положительной обратной связи, в которой различного рода информация циркулирует между участниками коммуникации, не обретая при этом законченной формы и порождая все новые и новые способы работы с информацией.

**1.1.8. Сети и протоколы**

От более общего понятия коммуникации следует перейти к конкретным техническим условиям, делающим коммуникацию возможной. Для этого мы обратимся к статье американского философа Александра Гэллоуэя «Сети» (Networks) из сборника «Ключевые термины теории медиа».

Рассматривая этимологию и значение этого слова в разных предметных областях: индустриальных системах, науках о жизни, некоторых разделах математики, социальных и гуманитарных науках, он подчеркивает два общих свойства для всех определений сети. Первая из них, - взаимосвязанность элементов сети – подразумевает, что только взаимодействие элементов составляет сеть, просто собранных в одном месте элементов будет недостаточно. Вторая – каждой сети присуща некоторая сложность.

Сеть состоит из «нодов», объектов сети, и «ребра», т.е. связи, которые соединяют объекты между собой. В зависимости от типа связей между нодами можно выделить три вида сетей: централизованные, децентрализованные и распределенная.

Централизованная сеть характеризуется одним центральным нодом, от которого исходят однонаправленные связи к другим, менее значимым точкам сети. Децентрализованная сеть состоит из нескольких нодов, связанных между собой, от которых отходят связи к другим точкам сети, но эти точки не связаны между собой. В распределенной сети нет центральных нодов, все они взаимосвязаны между собой.

В контексте интернет-коммуникаций сочетаются разные типы сетей, и то, как и по каким правилам осуществляется перемещение информации, определяют технологические стандарты, называемые протоколами: «Протокол – это набор рекомендаций и правил по применению технического стандарта, <…> системы материальной организации, структурирующие отношения битов информации и как они перемещаются по распределенным сетям». [Гэллоуэй 2010: 286]. По Гэллуэю, существование протоколов указывает на существование контроля в любой, даже распределенной сети, и нельзя противопоставлять новые коммуникационные сети уже существующим, таким как радио и телевидение: контроль существует везде, но его формы могут различаться.

Подведем итоги. Мы рассмотрели три функции медиа (обработка, хранение, передача) и основные термины, связанные с ними и необходимые для исследования. Далее мы рассмотрим понятие программного обеспечения (софтвера) как пространства, соединяющего в себе все медиа, существовавшие до этого (кино, фотография, фонография), и являющегося основной медиатехнической средой, с которой сегодня сталкивается любой субъект.

**1.2. Софтвер**

В работе с понятием софтвера мы будем опираться на книгу Льва Мановича «Софтвер берет на себя командование» («Software Takes Command»), поскольку в ней он описывает истоки появления определенного взгляда на компьютерные технологии, появление на смену идеи компьютера как сложного калькулятора, выполняющего расчеты для государственных и частных организаций идеи персонального компьютера как метамедиума с возможностью работы над культурными данными, переход от идеи софтвера как услуги и достоянием немногих программистов к софтверу как стандартизированному товару, производимому отдельным сегментом рынка труда.

Софтвер стал включать в себя все предыдущие способы обработки, хранения и передачи информации: «Я считаю, что софтвер пронизывает все аспекты современного общества. Таким образом, если мы хотим понять современные техники контроля, коммуникации, репрезентации, симуляции, теоретизации, принятия решений, памяти, видения, письма и интеракции наш анализ будет неполным без изучения софтвера». [Манович 2013: 13].

Обращение к теме софтвера продолжает работу Мановича по описанию языка новых медиа в одноименной книге, в которой он вводит понятие «культурного софтвера» и отмечает важность использования инструментария гуманитарных наук в изучении современного программного обеспечения.

Культурный софт делится на несколько категорий:

1. Софт для создания культурных объектов, содержащих идеи, репрезентации и мнения.

2. Софт для доступа, распространения и микширования уже существующих культурных объектов в Интернете.

3. Софт для создания и распространения информации и знания: редактирование страницы в Википедии.

4. Софт для коммуникации с другими людьми онлайн: электронная почта, социальные сети, видеочаты.

5. «Вовлекающий», интерактивный софт (например, компьютерные игры).

6. Софт, собирающий метаданные и другую информацию о пользователе (например, учитывающий предыдущие поисковые запросы или лайки в соцсетях).

7. Софт для разработки и поддержки уже работающих программ (например, программирование новых системных библиотек, написание других программ).

От основных видов программного обеспечения Манович переходит к истокам появления культурного софтвера. Для этого он обращается к текстам инженеров, создавших основные компьютерные технологии. В них они не только описывают технические спецификации той или иной технологии, но и более общие взгляды на то, как должны работать в технологии в обществе и каким может быть пользовательский опыт.

На формирование концепции метамедиума повлияли идеи когнитивного психолога Джерома Брунера. Он модифицировал идеи Жана Пиаже, который считал, что дети в своем интеллектуальном развитии проходят три стадии: кинестетическую, визуальную и символическую. Но в отличие от Пиаже, считавшего, что ребенок последовательно проходит все стадии, Брунер полагает, что когнитивные установки, присущие каждой стадии, сосуществуют и развиваются параллельно. Брунер дал немного другие названия для этих этапов: энактивная, иконическая и символическая. Хотя эти ментальности появляются на разных этапах развития психики, они продолжают сосуществовать. Кэй перенес теорию Брунера на компьютерный интерфейс: навигация и взаимодействие с объектами с помощью компьютерной мыши соотносится с энактивистской стадией, иконки и окна соотносятся с иконической стадией, поскольку в этот момент пользователь работает визуальной информацией, написание текста и кода соотносится с символической стадией. Графический пользовательский интерфейс задействует все когнитивные способности, и пользователь постоянно переключается между ними.

Опираясь на тексты Аллана Кэя и Адель Голдберг, одних из создателей ключевых технологий, на которых строится сегодня практически любое программное обеспечение, Манович делает вывод, что персональный компьютер задумывался не только метамедиумом (термин Кэя), содержащим в себе все предыдущие медиумы, но и добавит несуществующие до этого способы работы с культурными артефактами. По Кэю, компьютер должен стать «персональным техническим медиа», объединяющим в себе все предыдущие медиумы и добавляющие новые способы работы с ними. Важной составляющей этого метамедиума является созданный Кэем первый в мире язык объектно-ориентированного программирования, в теории позволяющий пользователям быстро создавать новые виды медиа при помощи уже существующих инструментов. Этот процесс происходит не сразу, но поэтапно. Манович так описывает этот переход внутри компьютерных технологий: «На первом этапе большая часть существующих медиа были смоделированы на компьютере, затем были созданы несколько новых видов медиумов, реализуемых только в цифровой среде. На втором этапе смоделированные и специфичные для компьютера медиа обмениваются способами работы и свойствами». [Там же: 39]. В этом случае следует отличать ремикс, т.е. комбинацию различных содержаний внутри одного медиума, от комбинации методов, техник и способов репрезентации, присущих новым медиа. Модернистская логика поиска своего уникального языка внутри медиума не подходит для персонального компьютера, поскольку он включает в себя все предыдущие медиумы и имеет бесконечный потенциал для их взаимодействия и появления новых функций и способов работы с ними. С точки зрения истории медиа, Манович выделяет два типа медиумов, которыми оперирует компьютер: симуляцию уже существующих медиа (текст, музыка, кино) и медиа, существующие исключительно в электронной форме и не имеющие аналоговых предшественников: трехмерное виртуальное пространство, гипертекст, и др. Также существует два типа работы с данными: «медиа-специфичные», т.е. применяемые только к одному определенному виду данных. Например, визуальные фильтры, применяемые для обработки фотографий в редакторе изображений, невозможно использовать для редактирования текста или видео. Другой тип – это инструменты, общие для всех типов данных. К таким относятся команды поиска, копирования, вставки, сортировки, связывание гиперссылками, и т.д. Репрезентация медиа-объектов и способов работы с ними дает пользователю иллюзию, что цифровые медиа симулируют аналоговые медиа и относится к ним стоит точно также. На самом деле, цифровые медиа симулируют техники создания, навигирования, редактирования и взаимодействия с медиаданными. Например, текстовый редактор предоставляет инструменты для написания и редактирования текста, изменения вида текста и просмотра финального результата. Эти инструменты он приравнивает к алгоритмам, а медиаобъекты к данным. Из этого факта Манович делает важный вывод о том, что ментальная модель пользователя повторяет структуру компьютерной программы, поскольку пользователь начинает мыслить в категориях возможных действий внутри программы и тех объектов и структур данных, на которые эти действия могут быть направлены (Манович замечает, что использует понятие структур данных не в том смысле, который подразумевается в информатике, но как обозначение видимую для пользователя организацию данных, с которой он может взаимодействовать).

Подводя итоги, Манович пишет, что развитие современных медиа стоит рассматривать не как набор отдельных медиумов с присущими им уникальными языками, но как экосистему, общую для всех видов и техник работы с медиаобъектами, основой которой является софт. Различные инструменты работы с данными могут использоваться совместно, составлять комбинации для дальнейшего использования и распространения. Все предыдущие термины теории медиа объединены в общей экосистеме софта, где существуют структуры данных для хранения информации, алгоритмы для ее обработки и протоколы для передачи. Именно благодаря софтверу и только в его окружении может существовать электронная литература.

**1.3. Электронная литература**

В этом разделе мы опишем само понятие э. литературы, его историю и особенности э. литературы как медиума, основные жанры, на которые подразделяются произведения, а затем рассмотрим жанр «code work», к которому относятся многие работы mez.

**1.3.1. Понятие э. литературы и ее особенности как отдельного медиума**

Для описания понятия электронной литературы будут использованы два обзорных труда по теме: «Электронная литература» Скотта Реттберга и «Электронная литература: новые горизонты литературы». Первая работа описывает краткую историю понятия, истоки электронной литературы, основные жанры и уже существующие тексты по теме. Книга Хэйлс сосредоточена на более общих философских и теоретических положениях, необходимых для интерпретации электронной литературы.

С самого начала важно уточнить, что электронная литература не является отдельным медиумом, жанром или просто набором работ, произведенных с помощью современных технологий. Также ее нельзя сводить к уже существующим родам литературы, таким как драма, лирика или эпос. Электронная литература – это «<…> культуная форма, коллективное создание терминов, ключевых слов, жанров, структур и институций, а также производство новых литературных объектов». [Табби 2011: 299].

История э. литературы связанна не только с появлением произведений, самые ранние из которых можно датировать 50-ми годами XX века, еще до появления персональных компьютеров, но и с появлением таких институций, как «Организация электронной литературы» (Electronic Literature Organization). Образованная в 1999 году, она поставила перед собой задачу как сохранения существующих текстов, так и распространения информации о новых произведения, создания программ курсов для университетов, посвященных э. литературе, и другой просветительской деятельности. Электронная литература во многом связана с институциями, создающими необходимые условия для ее сохранения и распространения.

Общее определение электронной литературы звучит так: «работа, имеющая выраженную литературную составляющую и использующая возможности и контексты, предоставляемые компьютерными сетями или системами». [Реттберг 2019: 15]. Одним из важных аспектов при восприятии электронной литературы являются не только герменевтические процедуры интерпретации значения, тем и языка работ, но и то, как читатель будет воспринимать софт, на котором запущено то или иное произведение: «Когда проект может быть представлен в виде компьютерной программы, напечатанного текста, инсталляции или перформанса, его восприятие меняется в зависимости от контекста с каждой новой итерацией». [Реттберг 2019: 48].

Здесь можно привести различие между компьютерными играми и другими медиа, которое использует исследователь компьютерных игр Маркку Эскелинен, поскольку оно также релевантно для выявления отличительных черт электронной литературы: «Обобщим: в искусстве нам, возможно, придется сконфигурировать, чтобы иметь возможность интерпретировать, тогда как в играх мы должны интерпретировать, чтобы иметь возможность конфигурировать и продвигаться от начала до выигрышной или какой-либо другой ситуации». [Эскелинен URL]. Вместо компьютерных игр в это определение можно поставить э. литературу не только потому, что многие произведения такого рода интерактивны, но и по причине медиатехнического окружения (текстовых редакторов, программ просмотра текстовых файлов и др.), которое тоже нужно сначала интерпретировать, понять, какие существуют возможности для читателя в данной программе, а затем конфигурировать, т.е. манипулировать доступными для читателя средствами при конструировании смыслов.

Одной из главных особенностей э. литературы Хэйлс считает вычисление: «Вычисление ни нечто второстепенное или случайное для э. литературы, но центральный аспект ее исполнения и интерпретации» [Хэйлс 2008: 31]. С этим аспектом она связывает особый режим работы с подобного рода текстами – они являются не четким набором правил, но постоянным взаимодействием на разных уровнях: интерпретации, замысла автора, его программного воплощения, актуализированных и неактуализированных возможностей и т.д. Причем эти возможности существуют как для автора, так и для читателя. Их пересечение и образует литературное произведение. Отличие такого подхода от рецептивной эстетики заключается в том, что произведение э. литературы активней откликается на действия читателя – не только текст влияет на восприятие читателя, но и действия читателя, сам процесс чтения влияет на текст. Как пишет Хэйлс, компьютер отменил монополию видения, ассоциированного с чтением – теперь тексты при взаимодействии с ними мобилизуют сенсомоторные реакции тела – нажатия на клавиши, управление компьютерной мышью, печать текста и т.д. Телесность становится полноправным участником в интерпретации текста.

Хэйлс также пишет, что э. литература создает новые цепи обратной связи между читателем и текстом: от сенсомоторных функций к порождению значений и от новых значений к сенсомоторным функциям. Э. литература является общим полем для взаимодействия программного кода и естественного языка, управляет той информацией, которую получает на входе и в зависимости от нее выдает некий результат.

Другой важный признак э. литературы – это «напряжение между значениями высокого уровня в человеческом дискурсе и многоуровневыми процессами, связанными с исполняемым кодом». [Там же: 75]. Таким образом, электронная литература актуализирует и исследует границу между операциями над знаками, составляющими синтаксические цепочки и исполняющими определенные команды, и значениями, присущими естественным языкам.

Теперь обратимся к визуальной стороне э. литературе, ее отображению на экране. Такие медиатеоретики, как К. Хэйлс и Л. Манович, пишут о двойной природе экрана – как плоской поверхности, на которой мы можем оперировать некоторыми объектами, и «окном в мир», создающим иллюзию трехмерного пространства. Также Хэйлс добавляет, что обычный текст создает иллюзию «авторского голоса», рассказывающего нам историю. При чтении любого текста с экрана такая иллюзия невозможна - от нас скрыт программный код, поддерживающий иллюзию чтения «книги», которая является всего лишь репрезентацией работы кода.

Э. литературу можно делить не только на традиционные жанры, т.е. не только в зависимости от восприятия читателя, но и от того программного софта или кода, на котором эта литература запускается или была написана. Реттберг обосновывает важность понятия жанра для классификации э. литературы тем, что «[Понятие] жанра дает исследователю основу для более точного изучения особенностей того или иного произведения. Жанр указывает на то, как читать, и, применительно к электронной литературе, где и что читать, давая возможность проникнуть дальше интерфейса и рассматривать содержание работы, находящееся за пределами ее формы и технических свойств». [Реттберг 2019: 20]. Границы между жанрами э. литературы довольно прозрачные, что приводит к гибридизации жанров и существовании текстов, использующих разные текстовые стратегии. Реттберг выделяет шесть основных жанров электронной литературы:

1. Гипертекстовая проза и поэзия – тексты, в основе которых лежит технология гипертекста – сети текстовых и визуальных фрагментов, соединенных ссылками и предполагающих нелинейное и многовариативное чтение.

2. Комбинаторная литература – к этому поджанру относятся компьютерные программы, генерирующие поэтические, прозаические и визуальные тексты.

3. Интерактивные, игроподобные формы – включают в себя большой спектр мультимедийных проектов, интерактивных произведений (иногда в форме компьютерной игры) и литературных приложений для смартфонов.

4. Кинетическая и интерактивная поэзия – такие работы родственны практике конкретной и визуальной поэзии, только вместо исследования материальности печатной страницы такого рода поэзия исследует возможности культурного софтвера.

5. Сетевое письмо (Network Writing) – использует возможности и существует в Интернете. Примером таких работ являются code work, распределенные нарративы, «fictional blogs», коллективные нарративы, импровизированные нарративы и т.д.

6. Тексты, не вписывающиеся в предыдущие жанры – к последней категории можно отнести другие поджанры, которые нельзя полностью включить в предыдущие пять категорий: локативные нарративы (предполагающие перемещение читателя в реальном пространстве), тексты с дополненной реальностью, литературные инсталляции, интерактивное кино и др.

Все многообразие поджанров э. литературы не исчерпывается этим списком, но в рамках данного исследования важно обратить внимание на сетевое письмо, т.е. использующее Интернет для написания и распространения текстов. Одним из подвидов сетевого письма является жанр «code work».

**1.3.2. Code work**

В данном разделе мы рассмотрим само понятие «code work», ключевые элементы, из которых он состоит, и возможные интерпретации и способы прочтения такого рода текстов исходя из тех или иных теоретических предпосылок.

Такое определение дает этому понятию Реттберг: «Code work подразумевает формы письма, исследующие двойственную природу цифрового создания текста с помощью компьютера и его существования в Интернете как в виде поверхностного текста, так и в виде кода. Хотя code work необязательно должны существовать в Интернете (некоторые работы предназначены для неподключенных к сети компьютерам), большая часть работ в жанре code work существует в Интернете». [Реттберг 2019: 180].

Понятие «code work» ввел художник и поэт Алан Сондхейм в одноименной статье. Это литературный текст, соединяющий в себе естественные и формализованные языки (языки программирования, гипертекстовой разметки и т.д.). Такой текст может читаться как человеком, так и компьютером (т.е. быть исполняемым). Но чаще всего данный гибрид не предназначен для чтения машиной. Сондхейм делит «code work» на три примерные категории:

1. Работы, которые смешивают синтаксис естественного языка с синтаксисом формализованных языков. К этому типу он относит работы mez, Талана Меммота, Теда Уорнелла и других авторов.

2. Работы, в которых «скрытый от глаз код изменяет текст на поверхности [интерфейса]». [Сондхейм URL]. Примером подобных текстов являются работы самого Сондхейма. На протяжении многих лет длится его проект «Internet Text», суть которого заключается в бесконечной переписке, которую ведет Сондхейм, отправляя на различные адреса свои тексты, большая часть из которых была изменена с помощью различных компьютерных программ.

3. Работы, в которых скрытый код и его исполнение становятся содержанием работы. Примером такого рода произведений является практика группы художников JODI. В их работах код часто изменяет работу протоколов Интернета и влияет на отображение веб страницы.

Одна из дискуссий, связанная с понятием code work-а – являются ли работы этого жанра исполняемым кодом – очень важна для нашего исследования. В программной статье 2002 года «Код это не текст (если только код не является текстом)» Джон Кэйли пишет, что большая часть работ в жанре «code work» не является кодом, поскольку они не адресованы машине и не могут быть исполнены. Другие авторы (например, Флориан Крамер, Дэннис Тэннон, Марк Марино и другие, о которых мы писали выше) не соглашаются с его доводами и считают, что культурный слой, присутствующий в программном коде, не менее важен, и нужно рассматривать не только как код взаимодействует на машинном уровне, т.е. исполняется, но и практики интерпретации кода человеком, которые не менее важны при написании кода и интерпретации пограничных явлений, где элементы кода инкорпорированы в тексты на естественном языке, в том числе и при чтении code-work-ов.

Электронная литература, и «code work» в частности, рефлексируют в определенной мере все вышеописанные способы концептуализации письма и текста в цифровую эпоху. С одной стороны, они пытаются с помощью различных средств уловить цепочку моментов перехода, когда некий знак все еще будто бы «скрыт» от человека, не функционален, ждет, когда им воспользуются, и его репрезентацией, которой манипулирует пользователь. В случае работ в жанре «code work» этот момент метафоризируется, но не преподносится в том виде, в котором он существует. С другой стороны, они исследуют возможности текста как интерфейса, где основное внимание переносится с внешних механизмов интеракции на малозаметные проявления скрытых за ними кодов.

**Выводы к первой главе**

В первой главе были описаны основные положения медиа-теории как области знания, изучающей техническое, опосредованное существование культурных артефактов, а также влияния их материальной основы на производство значения и их рецепцию сквозь призму теории литературы, философии, культурных исследований. Было выделено понятие информации как ключевое для современной ситуации, а также, в соответствии с теорией Ф. Киттлера, были выделены три способа работы с информацией, которые характеризуют современные медиа: обработка, хранение и передача. К каждому из них были подобраны соответствующие понятия из информатики, поскольку одним из главных медиумов на сегодняшний день является персональный компьютер, с его помощью создаются произведения э. литературы, в том числе и тексты Мэри Энн Бриз. Все эти термины будут использованы при исследовании наравне с уже существующими категориями теории литературы. Далее был рассмотрен феномен электронной литературы, историю его зарождения и институализации, а также один из поджанров, code work, в рамках которого созданы работы Мэри Энн Бриз, являющиеся объектом данного исследования.

# **Глава 2. Практика Мэри Энн Бриз в медиатехнической среде**

Во второй главе будут кратко охарактеризованы художественная и литературная практики mez, затем произведен обзор существующей литературы, посвященной ее творчеству. Иной способ интерпретации текстов mez будет начат с рассмотрения проблемы рецепции э. литературы в целом и ее текстов в частности, будет произведена критика интерпретации метода mez как «гибрида» нескольких языков. В этих рамках будет раскрыто подробней понятие «user amplification» как основного условия взаимодействия человека и компьютера. Затем будет рассмотрена проблема границы между чтением и письмом в текстах mez, где будет использовано иначе определенное понятие интерактивности, специфичное для нашего исследования. После этого с помощью терминов теории систем и кибернетики второго порядка будут исследованы методы производства текстов, которые мы отнесем к системе амплификации, связанные с программным кодом и взаимодействием пользователя с графическим интерфейсом, и системам означивания, во взаимодействии которых появляются литературные тексты. После этого мы рассмотрим, как с помощью текстов mez изменяются отношения между откладыванием значения и амплификацией пользователя. В последней части второй главы будут рассмотрены понятия эксплойта, текстового вируса и экстенсивного использования языка, с помощью которых мы пересмотрим устоявшуюся оппозицию исполняемого и неисполняемого. В дополнение к существующим понятиям мы рассмотрим дополнительные термины выполнения (performance) и исполнения (execution), которые помогут пересмотреть практику mez в современном медиатехническом контексте, отличном от существовавшего во время создания этих текстов.

**2.1. Художественная и литературная практики Мэри Энн Бриз**

Первые работы Мэри Энн Бриз появились в 1993 году, когда она начала использовать Интернет для создания и распространения своих текстов. На сегодняшний день она является автором множества работ в жанре цифрового искусства: VR-инсталляций, интерактивных нарративов, экспериментальных компьютерных игр и других произведений. Но центральным для данного исследование будет ее проект Mezangelle – язык, состоящий из элементов естественных и формализованных языков. Так его описывает сама авторка: «С середины 90-ых годов Mezангелле постепенно формировался в e-mail переписке, языках программирования, чат-приложениях. Целью было соединить английский язык, поэтические конвенции, программный код, социальный комментарий и онлайн коммуникации». [Антология нетарта URL].

Тексты mez в определенном смысле стали «каноном» электронной литературы, одним из примеров того, как можно писать и распространять литературные тексты в современном медиатехническом окружении, и каким образом меняется рецепция текста на экране компьютера. О ее произведениях написано множество научных работ, и за тридцать лет существования ее «code work» проекта «mezangelle» сложился определенный консенсус насчет того, как эти тексты следует интерпретировать и читать, что они говорят нам об изменениях в восприятии создания и рецепции литературы. В следующем разделе будут рассмотрены устоявшиеся интерпретации текстов mez от различных литературоведов и медиатеоретиков.

**2.1.1 Устоявшиеся интерпретации текстов mez**

На данный момент существует устойчивый способ интерпретации текстов mez и их вписанности в определенную традицию. Ниже мы рассмотрим несколько примеров подобных интерпретаций, чтобы выявить их общие черты и ту концептуальную рамку, которую использую исследователи.

Кэтрин Хэйлс описывает тексты mez как «билингвальную практику, которая разрушает общепринятую связь между фонемой и письменным знаком и формирует новые связи между кодом и английским языком». [Хэйлс URL]. Для нее тексты mez представляют из себя пиджин, смешивающий языки людей и машин и подрывающий господство языков программирования в медиатехнической среде.

Рита Рейли в статье «Интерференции» предложила одну из устойчивых интерпретаций практики mez как создания своего собственного гибридного языка и обнажение скрытых от глаз пользователя процессов, происходящих в компьютере. На примере работ коллектива Jodi она вводит важное различие для критического исследования подобных текстов между «кодом, обладающим глубиной и операциональностью, и код, изолированный на поверхности текста». [Рэйли URL]. На уровне лексики она выделяет такие особенности, как использование разговорного стиля, присущего переписке в чате, для которой характерна замена морфем и предлогов цифрами («2 4m a text»), неконвенциональное использование пунктуационных знаков: точка не разделяет предложения, но соединяет синтагмы в значимые группы, задает иерархию, как в адресации веб-страниц, где точка отделяет доменные имена разных уровней, или для отделения объекта от его свойств, как в объектно-ориентированном программировании. Аспект интерактивности также исследуется Рэйли – она пишет, что mez отправляет свои тексты по электронной почте, а затем включает ответы на него в первоначальный текст. Mezangelle функционирует как «модус поэтической композиции и обыденной (general) коммуникации». [Рэйли: Там же].

Немецкий медиатеоретик Флориан Крамер в статьях «Концепты, системы записи, программное обеспечение, искусство», «Поэзия программного кода», «Цифровой код и литературный текст» и «mez, \_Виро.Логическое Обусловли] [вание] [1.1\_ Анализ Текста» рассматривает тексты mez различными способами: от медленного чтения отдельного текста в статье Крамера до вписывания ее практики в более широкий контекст электронной литературы, проблем исполнения и соотношения текста на экране с невидимым для пользователя кодом. В своих текстах о mez и о других авторах подобных работ Крамер подчеркивает не потенциальную исполняемость или неисполняемость подобных работ, но культурные импликации программного кода, исследуемые текстами mez. Крамер пишет, что «На самом деле цифровых медиа не существует, есть только цифровая информация, становящаяся «медиа» только с помощью аналоговых средств вывода: мониторов, динамиков, принтеров. Подобного рода конверсия от цифрового к аналоговому возможна благодаря аппаратному обеспечению: видеокартам, звуковым картам или технологиями последовательной передачи данных (serial interfaces)». [Крамер 2013: 69]. Если основой цифровой информации является программный код, а алфавит является одним из способов кодирования информации и используется в языках программирования высокого уровня, то тексты жанра «code work» работают, согласно Крамеру, с минимально возможным опосредованием между кодом и естественным языком. Произведения жанра «code work» акцентирует внимание не на поверхности интерфейса, но на лежащем в основе коде, соединяющем в себе два ключевых для лингвистики концепта – структуру языка, разрабатываемую структурализмом, и прагматику, которой занимается теория речевых актов.

В рецензии на книгу «Human Readable Messages» Роб Майерс пишет, что тексты mez «скорее воссоздают (recreates) экспрессивные, коммуникативные основы софтвера, а не просто эстетику его поверхности». [Майерс URL]. Кроме этого, он подчеркивает особенности чтения этих текстов, отличных от обычного медленного чтения, организующего интерпретацию: «Читать [тексты] Mezangelle значит их парсить. Парсинг – это деление компьютером текстовых входящих данных на небольшие фрагменты значимой информации». [Там же]

В другой рецензии на эту книгу Энди Карратерс кроме перечисления уже известных в работах других авторов тезисов о текстах mez добавляет, что ее «поэзию» можно рассматривать в постструктуралистском ключе. Поскольку современная постмедийная ситуация схватывается в цифровой репрезентации, выраженной программным кодом, любого предшествующего медиума, будь то фотография, кино или музыка, она совпадает с тем, как постструктурализм описывает современное состояние культуры как лингвоцентричной, в которой любой культурный артефакт интерпретируется с помощью понятия Текста как общего пространства означивания, в котором границы между отдельными произведениями и жанрами размыты. Таким же образом и тексты mez избегают однозначного определения, существуют медиапространстве, опосредованном программным кодом.

Салли Эванс в эссе «Оружие против Логоса: избыток значения и субъективность в поэзии Mezangelle» описывает ее тексты как «лиминальную точку между смыслом и бессмысленностью, логикой и алогичностью» [Эванс URL]. Она указывает на множественные модели субъекта, используемые mez для написания и распространения текстов, и работу с материальностью языка, обнаруживающую себя в сложности интеракции между человеком и компьютером. На множественность различных субъективных позиций также указывает Джон Рип в своей статье «Факт в том, что я фикция: Мэри Энн Бриз, ее аватары и трансформация субъективности». [Рип URL].

Илья Шиляк с помощью понятия «малой литературы» Делеза и Гваттари рассматривает тексты mez как практику создания «языка меньшинств», в котором смешиваются доминирующие человеческие и машинные языки, при этом работа по означиванию и исполнению в «креольском языке, подрывающем предписанную утилитарность слова и кода» [Шиляк URL] приостанавливается.

Отдельно стоит выделить статьи исследователя и практика э. литературы Джона Кэйли, поскольку он уточняет понятие исполнения применительно к э. литературе, и от критики его исследований, проведенной другими авторами, будет отталкиваться наша интерпретация. Статьи Кэйли о границе программного кода и текста и о применении понятия кода к произведениям жанра «code work» вызвали полемику и множество откликов в сообществе.

В статье «Код – это не текст (кроме случаев, когда он является текстом)» он пытается разграничить понятия текста и кода, которые, по его мнению, другие авторы смешивали. Для него ключевым критерием является адресуемость кода машинным процессам – если «псевдокод», такой, как в работах mez, не исполняем на компьютере, то он является (литературным) текстом, адресуемым человеку, и к нему нельзя применять термин код. Такого рода тексты он называет «интерфейсными текстами» (interface texts). Далее он критикует устоявшееся описание смешивания естественных и формализованных языков в работах жанра «code work» как вскрытие, обнажение невидимых пользователю структур программного кода, поддерживающим иллюзию прозрачного интерфейса. Для него такого общего описания недостаточно, и он указывает на те аспекты, которые работы в жанре «code work» не актуализируют: «Что [программный код] говорит человеку, не разбирающемуся в программировании о свойствах и власти кода? Можно ли это все еще называть кодом? На каком уровне в цепи мерцающих (flickering) означающих он находится? Отражена ли его функция в иерархии кодов при включении в «интерфейсный текст»?». [Кэйли 2018: 59]. Также он критикует метафорическое применение к «code work» терминов пиджин и креольский язык, поскольку такого рода гибридные тексты не формируют полноценных языков. Пиджины формируются при смешении двух и более полноценных языков, а не при скрещивании находящихся на разных ступенях иерархии символических системах, адресуемых разным сущностям. Главный аргумент Кэйли заключается в том, что литературная работа только тогда может называться кодом, когда она на одном уровне читается как литературное произведение и адресуемо человеку, а на другом уровне является программным кодом, адресуемым машине и изменяющим ее работу: «Код, не являющийся текстом, рабочий и не фрагментированный, может порождать, в определенный промежуток времени, текстовые стратегии означивания». [Там же: 96].

В статье «Время, код, язык: поэтика новых медиа и программируемое означивание» он продолжает тему различия кода и текста, начатую в статье «Код — это не текст». Он критикует классификацию работ в жанре code work, предложенную Аланом Сондхеймом за то, что он опирается на поверхностную репрезентацию кода в интерфейсе, а не на то, каким образом код работает и изменяет состояния компьютерных процессов. Он вводит свою классификацию:

1. Код как язык.

2. Язык с элементами программного кода (код неисполняем)

3. Код с элементами естественного языка (код исполняем, но слова естественного языка нечитаемы)

4. Код как система соответствий, кодирование

5. Код как работающая программа, производящая тексты или необходимая для их производства.

Первые три категории охватывают работы жанра «code work», которые, согласно определению Кэйли, неисполняемы на компьютере. Четвертая категория соответствует функции кодирования, одному из главных свойств языка программирования любого уровня. К пятой категории Кэйли относит непосредственно те работы, которые представляют из себя рабочие программы, запускаемые на компьютере и имеющие признаки литературных текстов: например, генераторы текстов или интерактивные, мультимедийнфе работы. Далее он вводит понятие «паратекстуальное программирование». Поскольку программирование, в определенной степени, можно назвать любое действие, совершаемое пользователем (меняется лишь уровень абстракции), к паратекстуальному программированию можно отнести действия с текстом на поверхности интерфейса: например, работу в текстовом редакторе. Работы mez и других представителей жанра Кэйли относит именно к паратекстуальному программированию, и одним из ее важных аспектов является «постмодернистская пунктуация» (термин Кэйли). К этому понятию он относит неконвенциональное использование пунктуации, подражающее пунктуации языков программирования. Он утверждает, что подобного рода пунктуация характерна не только для языков программирования, но и активно используется во многих модернистских и постмодернистских текстах. Вынесенная из контекста кода исполняемой программы, такого рода пунктуация становится еще одним средством выражения фрагментированности и сложности цифрового мира, но не перенесением средств выражения из одного языка в другой.

В целом, Кэйли старается отграничить культурные аспекты, существующие в коде, и дать четкие критерии того, в каких случаях стратегии означивания, присущие литературе, взаимодействующие с кодом могут считаться электронной литературой, а какие просто смешивают различные языки на поверхности интерфейса.

Подводя итоги, можно сказать, что общим местом в интерпретации произведений mez является указание на смешение естественного и формализованных языков в текстах, которое обнажает скрытые от пользователя информационные структуры, генерирующие видимое на экране и задающие способы взаимодействия с информацией. Это позволяет в дальнейшем интерпретировать себя в цифровом окружении по-новому. Также исследователи подчеркивают визуальную составляющую текстов mez и то, как она влияет на их чтение. Ключевой аспект ее работ, – потенциальная исполняемость или неисполняемость текстов, будет рассмотрен через понятие интерактивности, которая является результатом исполнения программного кода и считывается на уровне пользователя. От уровня языков программирования высокого уровня, на котором акцентирует внимание Кэйли, следует перейти к уровню пользовательского взаимодействия с графическим интерфейсом, которое не менее важно, чем работа с программным кодом.

# **2.2 Рецепция текстов mez в современном медиатехническом окружении. Критика интерпретации текстов mez как языка-пиджина**

# **2.2.1 Применение понятия интерактивности в интерпретации электронной литературы и текстов mez. «User amplification» как основное условие взаимодействия человека и компьютера**

С середины XX века тема читательской рецепции литературных текстов является одной из самых обсуждаемых в литературной теории. Современные технологии кардинально меняют способы чтения литературных текстов, и эта ситуация рефлексируется в том числе авторами э. литературы, создающей новые формы взаимодействия читателя с текстом, причем многие из них подразумевают прямое влияние на развертывание (в случае гипертекстовой литературы) и содержание (в случае интерактивных форм) произведений. Поэтому для начала мы рассмотрим присущее, по крайней мере на первый взгляд, э. литературе понятие интерактивности, после этого изучим некоторые способы применения понятия чтения к текстам mez, в которых нет явных интерактивных элементов.

Понятие интерактивности используется во множестве контекстов – от компьютерных игр, создатели которых выделяют интерактивность как отличающую особенность их медиума, до science art, media art, netart и других направлений современного искусства, работающих с современными технологиями. Довольно рано это понятие стали критиковать многие теоретики и художники. Флориан Крамер пишет о том, что расхожее понятие интерактивности взяло за основу кибернетический процесс обратной связи, который основан на бихеоверистской теории взаимодействия субъекта с окружением посредством реакции на внешний стимул. Поскольку все возможные виды реакции предзаданны разработчиком, Крамер считает, что в техническом объекте не существует настоящей интеракции, но это не значит, что не существуют интерактивные технологии и интерактивный дизайн, регулирующие возможные способы взаимодействия человека и окружения. К ним он относит архитектуру и язык.

Например, в 1996 году медиахудожник Алексей Шульгин в манифесте «Искусство, власть и коммуникация» критикует ситуацию, сложившуюся в современном искусстве новых медиа на тот момент, когда понятие интерактивности сужалось до простой манипуляции, выбору между несколькими предзаданными, мало чем отличающимися вариантами. В его представлении, медиа искусство ознаменовало собой переход от репрезентации к манипуляции. Он связывает такого рода понимание интерактивности со слишком сильной зависимостью художников от медиума и предлагает им более критически подходить к цифровым технологиям при проектировании работ.

Исследовательница цифрового искусства Кристиана Пол пишет об обратной ситуации, когда под термином интерактивность понимают все более широкий спектр взаимодействий реципиента и произведения искусства. Также Пол упоминает интерактивность в качестве внутреннего процесса, присущего восприятию визуального искусства. Для нашего исследования подобная интроицированная интерактивность очень важна, поскольку при акценте на цифровую интерактивность, где пользователю эксплицитно представлены способы работы с интерфейсом и возможные результаты его деятельности, понятие интерактивности упрощается и привязывается к определенному контексту (например, исследованиям компьютерных игр или полю цифрового искусства). Важно проследить, как процесс интеракции выходит за пределы того, что В. Чан и Л. Манович называют «user amplification», т.е. репрезентации причинности между действием и результатом на поверхности интерфейса, разворачиванием нарратива из простой пользовательской манипуляции, присущих компьютерным играм и цифровому искусству. Как одна строка кода языка программирования высокого уровня равна нескольким строкам машинного кода, так и одно действие пользователя в интерфейсе может приводить к нескольким результатам, при этом для обоих случаев характерно отсутствие видимого разрыва между командой и ее выполнением, которое представляется пользователю и программисту в виде моментальной видимой реакции на его действие.

Вэнди Чан пишет о том, что с «user amplification» связано удовольствие от программирования, управления «микромиром», внутренне непротиворечивый в отличие от «внешнего» мира: «в то время, как другие инструменты воздействуют на внешний мир и являются всего лишь его частью, компьютер содержит свой мир в миниатюре <…> в котором программист обладает абсолютной властью». [Чан 2011: 46]. Для нас важно подчеркнуть именно момент закрытости, ложной отделенности двух миров.

Подобная закрытость подчеркивается не только на уровне кода, но и на уровне интерфейса, спроектированного таким образом, чтобы давать пользователю иллюзию совершения выбора, хотя все возможные варианты выбора уже спроектированы, и сам интерфейс устроен таким образом, что сам «ведет» пользователя к определенным действиям. Такого рода концептуализация приводит к ошибочному суждению о том, что «мир, как компьютер, заключает в себе невидимые принципы, которые мы можем отследить по их визуальным репрезентациям. Отсюда вытекает популярность софтвера как метафоры для почти всего – культуры, генетики, жизни, - и редукции всего <…> к процессам ясной коммуникации». [Чан 2011: 91].

В другой статье она цитирует ученого Дэвида Эка, который описывает создание языка программирования как проектирование виртуальной машины, но без ограничений, накладываемых аппаратным обеспечением. Таким образом, «user amplification» поддерживает неразрывность пользовательского опыта, связь между командой и ее исполнением. Такого рода неразрывность связана с прагматическим аспектом использования компьютера для выполнения практических задач. Многие произведения э. литературы «остраняют» прагматическое использование компьютера, нарушают привычные паттерны человеко-машинного взаимодействия, рефлексируют по поводу читательского опыта и его трансформации, вводя интерактивные элементы или элементы других медиумов. Активная роль читателя не только в интерпретации, но и в изменении произведений э. литературы заставляет рассмотреть вопрос границы между чтением и письмом применимо к текстам mez.

**2.2.2 Проблема границ чтения и письма в текстах mez.**

Проблема диффузной границы между чтением и письмом была поставлена еще постструктуралистами Ж. Деррида и Полем де Маном. Для них текст всегда уже является составленным автором произведением из других текстов. В определенном смысле автор тоже является читателем написанных до него текстов, поэтому встает вопрос – где проходит граница между читателем и писателем? Ведь если читательская интерпретация тоже представляет из себя своего рода текст, который отличается при каждом новом чтении и потенциально может быть записан, то разница между чтением и письмом практически полностью стирается: «различение автора и читателя — это одно из тех различений, ложность которых становится вполне очевидной по ходу чтения. Деконструкция — это не наша добавка к тексту, но как раз то, что в первую очередь и конституирует текст. Литературный текст одновременно утверждает и отрицает авторитет своего собственного риторического модуса, и, читая текст так, как мы его читаем, мы всего лишь пытаемся приобрести статус столь же строгого читателя, сколь строгим должен быть автор для того, в первую очередь, чтобы высказаться». [де Ман 1999: 27].

В контексте электронной литературы, особенно в отношении текстов mez, нужно учитывать иные способы чтения, например, чтение программного кода программистом. Марк Марино описывает, как деформируется привычный процесс чтения, когда его объектом становится код: «литературной грамотности <…> недостаточно для чтения кода. <…> Чтение кода требует установления адекватного понимания состояния софтвера (иногда и хардвера) в любой момент времени работы программы». [Марино URL].

Кроме изменения процесса чтения письмо так же претерпевает трансформации. Например, Дэннис Тэннон пишет, что если письмо на бумаге не опосредованно чем-то иным, то между нажатием клавиши и появлением текста на экране происходит несколько операций, разнесенных во времени и пространстве, и то, что мы видим на экране, не совпадает с внутренними процессами вычисления: «Например, стирание слова с бумаги подразумевает физический процесс удаления чернил. Наоборот, при «цифровой» симуляции стирания после нажатия клавиши, удаление с экрана не означает удаления данных с жесткого диска. «Стертое» слово может остаться или даже копироваться на другие носители. Стирание в этом случае больше не уничтожает, а скорее свидетельствует». [Тэннон 2017: 14].

В своей книге «Программируемое видение» Вэнди Чан актуализирует проблему связанности письма и чтения в связи со спецификой работы компьютера. Она пишет, что «чтение – это письмо в другом месте», имея ввиду процесс считывания информации с носителей и записи уже обновившихся данных в другие ячейки памяти. Эти два процесса практически неразличимы для человека, поскольку совершаются с очень большой скоростью, недоступной для человеческого восприятия.

От процессов, локализованных на более низких уровнях абстракции написания и исполнения кода перейдем к поверхности интерфейса, с которой взаимодействует обычный пользователь. Кэтрин Хэйлс в статье «Как мы читаем» описывает три вида чтения, актуальные на сегодняшний момент: медленное чтение, гиперчтение и машинное чтение. К первому из них относится внимательное чтение литературного текста, заостряющее внимание на стиле и языке работы и детально разбирающее его языковые особенности. Гиперчтение описывает считывание текста с экрана компьютера. Оно нелинейно и включает в себя множество различных паттернов взаимодействия скорее с информацией, а не со значением, как во время медленного чтения. К видам гиперчтения можно отнести поиск по ключевому слову, использование поисковика, работа одновременно с несколькими открытыми окнами на экране, копирование и вставка текста, перемещение по гиперссылкам и т.д. К машинному чтению относится использование алгоритмов для анализа различных паттернов в больших корпусах текстов, которые невозможно прочесть человеку. К такого рода алгоритмам относятся анализаторы частотности слов, алгоритмы тематического и стилистического анализа и др. Хэйлс делает вывод, что машинное чтение и гиперчтение делают ставку на распознавание паттернов, а значение и контекст занимают второстепенную роль, тогда как в процессе медленного чтения наоборот значение и контекст выходят на первый план, а распознавание паттернов играет зависимую роль.

Если по отношению к литературе постмодернизма можно сказать, что в процессе рецепции читатель конструирует свой собственный текст из предоставленного автором текстового материала, но это конструирование происходит в сознании читателя, то в случае э. литературы этот процесс овнешняется и буквализируется. Читатель напрямую манипулирует фрагментами текста в графическом интерфейсе произведения.

Тексты mez находятся в промежуточном положении между этими случаями. С одной стороны, они принадлежат к э. литературе и не могут восприниматься вне того медиатехнического окружения (протоколы электронной почты, почтовые программы, текстовые редакторы, PDF-файлы), в котором существуют. Ее тексты также требуют вовлеченного чтения, как и постмодернистские произведения второй половины XX века. С другой стороны, их нельзя назвать интерактивными в том узком смысле, сведенном до манипуляции предзаданными автором элементами. Как уже было упомянуто в предыдущем разделе, тексты mez считаются «гибридами», смесью английского языка и различных языков программирования. Исходя из этого понимания авторы работ о mez выстраивают стратегии чтения ее текстов. Но природа этой гибридности довольно проблематична. Вместо устоявшегося понимания гибридности как смешения содержания двух языков, мы будем использовать понятие «медиа-гибридности» (media hybrid), которое ввел Лев Манович в книге «Софтвер берет на себя командование». Под медиа-гибридностью он понимает неразрывное сосуществование в одном пространстве техник и методов, присущих разным медиумам, ставшими элементами программного обеспечения, а также абсолютно новые способы работы с данными, которые появляются при гибридизации. Например, в анимационном дизайне с текстом можно взаимодействовать с помощью методов, присущих производству кино и компьютерной анимации. Он отличает медиа-гибридность от простого сосуществования разных медиумов на веб странице или других мультимедийных документах. Мультимедиа не устраняют автономию разных медиа, они сохраняют свои способы организации данных, их модификации и доступа к ним. Также он отличает медиа-гибридность от конвергенции, где разница между различными медиумами при их скрещивании становится меньше или совсем исчезает.

Если применить это понятие к текстам mez, то становится ясно, что интерпретация ее практики как смешения нескольких разнородных языков имеет ввиду сочетание содержаний этих языков, т.е. того, что Кэйли называет интерфейсным текстом. При этом данная трактовка не учитывает техники и методы, а также софтвер, с помощью которых производятся литературные тексты или программный код, актуализируемые в практике mez.

С помощью понятия медиа-гибридности можно рассмотреть, какие способы означивания mez берет из литературы и из программного обеспечения, используемого для создания и распространения текстов, а также какие методы производства текста она берет из программирования и литературы. Для этого будут использованы понятия теории систем и кибернетики второго порядка, поскольку традиционной методологии литературоведения недостаточно для изучения медиатехнического окружения текстов mez и того влияния, которое оно оказывает на реципиента.

**2.2.3 Применение терминов теории систем и кибернетики второго порядка к текстам mez**

В этом разделе будут использованы работы исследователя литературы и науки Брюса Кларка, поскольку он рассматривает литературные тексты с точки зрения теории систем и кибернетики второго порядка.

Для начала дадим определение понятию системы: «[Система – это] сложное составное целое, объединенное таким образом, что порождение процесса происходит только при зависящих друг от друга взаимодействиях элементов [этого целого]». [Кларк 2014: 1]. При чтении литературного текста сталкиваются две системы: литературный текст как система знаков и человек, включающий в себя множество систем, например, психические системы, продуцирующие восприятие и знания о тексте, социальные системы, заключающие произведение в определенный литературный контекст, и т.д. Все системы можно разделить на аутопоэтические и неаутопоэтические. Первые способны на полную автономию – т.е. способны к порождению значения, понимания того, что они делают и как взаимодействуют с окружением. Процесс порождения значения цикличен, система такого рода всегда возвращается к себе и пересматривает уже существующие данные. Неаутопоэтические системы, несмотря на разную степень автоматизации тех или иных процессов, должны быть частью других систем, способных производить значения. В таких системах получение и обработка информации происходит линейно, она может находиться только в одном состоянии из двух возможных: открытом или закрытом.

Технические объекты неаутопоэтические, но при этом занимают двойственную позицию – с одной стороны, они не могут производить значение, но при этом они передают значения, являются экстернализованными хранилищами для коммуникационных и когнитивных операций, присоединенных к аутопоэтическим системам. Технические системы участвуют в аутопоэзисе, но не обладают им. К такого рода системам Кларк относит нарратив и письмо: «Текст напоминает фрактальную границу (fractal border) или интерфейс. Он одновременно и знаковая структура – материальный объект, структурный элемент, тесно связанный с самопродуцирующим аутопоэзисом социально-системной коммуникации <…>, и одна из форм дискурса, связанная с нарративной агентностью, литературным или кинематографическим повествованием». [Там же: 17-18].

Система существует в определенном окружении и взаимодействует с ним, обрабатывает информацию, полученную через разные (перцептивные) каналы, и действует согласно обработанным данным. Окружение всегда сложнее и разнообразнее системы, поэтому система вписывает в себя отличия между ней и окружением, и на основе этих различий создает внутри себя определенную область внешних отсылок, которая является для этой системы окружением. Система может включать в себя несколько подсистем, и тогда при коммуникации между ними данные переводятся на язык получающей данные подсистемы. Познание мира субъектом возможно лишь при конструировании возможностей различий между ним и миром, в процессе которых создаются формы восприятия окружения.

Другое важное понятие, которое используется в теории систем – это операциональная закрытость (operational closure). Оно означает определенную автономию системы от среды – у системы есть фильтры, которые отделяют необходимые для системы данные от ненужных.

Если применить вышеописанные термины к произведениям mez из книги «Human Readable Messages», то можно сказать, что субъект ее литературных практик – это система, взаимодействующая с медиатехническим окружением – не только происходящим на экране, но и миром в целом, опосредованным технологиями. Искусственный язык mezangelle состоит из двух подсистем, каждая из которых характеризуется своими техниками работы с элементами различных языков. Мы попробуем реконструировать те методы, которая она берет из производства литературных текстов и способы создания программного кода. На «выходе» получается нечто третье, обладающее уникальными свойствами и способами работы на границе этих двух языков, а не просто чередование языковых систем на различных, от фонетики до синтаксиса, уровнях языка, принуждающих читателя переключать внимание со способа означивания, присущего английскому языку на способ организации знаков, существующий в языках программирования. В текстах mez разрабатываются новые способы взаимодействия между амплификацией пользователя, присущей компьютерным системам, и тем, что в первом приближении можно назвать постоянным «нарушением ожиданий», несовпадением между установками читателя и читаемым текстом. Эти разнообразные феномены, которые для наглядности противопоставления мы сведем в один, описаны многими литературоведами и философами XX века, от В. Шкловского и его понятия «остранения» до проблематики письма у Деррида, несовпадения «горизонта» смыслов читателя и произведения в герменевтике и рецептивной эстетике, и др. Иначе говоря, в текстах mez разрабатывается взаимодействие между сложной синтаксической структурой языков программирования, которая оперирует потоками информации, не самореферентными и не обладающими значением, и литературной системой, оперирующей различными стратегиями означивания.

Далее мы отдельно рассмотрим вышеназванные стратегии, а затем вернемся к вопросу о формах чтения, возникающих благодаря медиа-гибриду систем амплификации и систем означивания.

**2.2.4 Системы амплификации и системы означивания**

Для начала, рассмотрим то, какие элементы языков программирования использует mez. Вот как в одном из литературных текстов исследуемой книги она описывает то, что использует в своей практике: «фрагменты языков программирования, в основном Perl, форматы [организации текста] операционных систем и электронной переписки, древовидные файловые структуры, <…> символьную запись булевой логики, команды, используемые в командной оболочке [семейства операционных систем] Unix, конвенции HTML, [языка программирования] Java и [кодировки] ASCII, а также обработанный (parsed) поэтический язык с аллюзиями на гиперссылочную коммуникацию посредством заключения в скобки значений и фрагментации слов». [Бриз 2011: 31]. Другие авторы, такие как Флориан Крамер и Рита Рэйли выделяют также использование пунктуации, репрезентирующее отношение между символами в языках программирования: использование нижнего подчеркивания как обозначения данных и переменных, а также для создания дистанции между различными элементами, квадратных скобок в нескольких значениях: определенного типа данных – списка, обмена сообщениями, ссылками на отдельные элементы структур данных, для группирования отдельных синтагм и морфем для смещения их значений. Можно сказать, что в основном она использует элементы и структуры программного кода, регулирующие его синтаксис, задающие линейную последовательность чтения, присущую программному коду, а не литературному тексту, побуждающему возвращаться к уже прочитанному и постоянно пересматривать уже известное. Также исходя из текстов (особенно из инкорпорированных фрагментов чужих текстов) можно увидеть, что она использует стандартные инструменты любого текстового редактора, такие как выделить/копировать/вставить. Но нужно отказаться от расхожей интерпретации такого рода соположения различных текстов как одного из вариантов коллажной техники, метода нарезок или схожих практик дадаистов, леттристов и сюрреалистов. Вышеназванные техники основаны на сочетании разнородных материалов, сталкивающихся на одной поверхности, и из этого столкновения получается эстетический эффект. Такого рода процесс нельзя повернуть вспять – если отклеить часть коллажа, то может получится другое произведение, но поверхность не вернется в предыдущее состояние.

Также нельзя назвать подобного рода практики, встречающиеся в редакторах изображений, простым воспроизведением существующих способов работы с изображением или текстом. Например, нельзя рассматривать pdf файл с книгой как простую копию бумажной книги, и уделять внимание только переносу текстового содержания без исследования специфичности софтвера, на котором запускается книга, поскольку в этом случае мы бы подверглись уже описанной нами иллюзии идеальной коммуникации между медиумами, при котором главную роль играет «содержание» того, что передается (текст книги), а сами условия передачи и хранения остаются за скобками. Такого рода понимание сохраняется и у некоторых медиатеоретиков. Например, в понятии транскодинга Льва Мановича.

Вэнди Чан указывает на некоторые недостатки этого понятия: «понятие транскодинга Мановича концентрируется на статичных данных и понимает вычисление лишь как перевод. Возможность программирования означает не только возможность по-новому манипулировать образами; это также означает, что компьютер все время совершает действия, которые мы не можем контролировать. Понимание софтвера лишь как перевода стирает необходимость вычисления для работы компьютера. Двойственной процесс чтения, происходящий в компьютере, не просто транскодирует или переводит код в текст/изображение/звук, или наоборот. Это чтение – в котором смешиваются чтение и письмо (для компьютера, чтение – это письмо в другом месте) – также принимает участие в других невидимых чтениях». [Чан 2011: 91]. Не только мы «читаем» книгу с экрана компьютера, но и он «считывает» наши действия: расположение курсора, полоски прокрутки, номер страницы, на которой мы находимся и другие данные собираются компьютером и могут использоваться другими людьми – например, дизайнерами пользовательского опыта, постоянно улучшающими интерфейс программ, или программистами, устраняющими ошибки в коде.

Электронные тексты mez не являются коллажами, поскольку поверхность интерфейса не меняется от прибавления или удаления различных элементов. В компьютере вся информация представлена в числовом виде, она однородна, поэтому к ней нельзя применить понятие коллажа, если говорить о процессах, происходящих на нижних уровнях абстракции. Текст на поверхности интерфейса можно перезаписать бесконечное количество раз без каких-либо последствий для «страницы», существующей в текстовом редакторе. Чтобы детальней рассмотреть это противопоставление, мы воспользуемся пониманием процесса, присущим системной теории. Вместо оппозиции статики и динамики в теории систем обратимой, устойчивой структуре, содержащей в себе ограниченный набор вариантов для выбора противопоставляется процесс как нечто необратимое во времени, обладающее иным качеством на каждой стадии своего разворачивания. Текстовый редактор и письмо в нем статично, поскольку документ можно подвергнуть бесконечному количеству изменений без влияния на работу самой программы. Создание текста в таком редакторе трудно соотнести с практикой письма, осмысленной при других медиатехнических условиях. Отличие между ними заключается в том, что при работе в текстовом редакторе исчезает не-информационное измерение текста, поскольку каждый символ, каждый отступ закодированы заранее, и на уровне пользовательского интерфейса они существуют в виде постоянно обновляющегося изображения, петли отрицательной обратной связи, а на более низших являются комбинациями двоичных значений. Пользователь, он же автор текстов, работает только с графическим интерфейсом и основными командами, доступными ему. Он перемещается в пространстве амплификаций, дающих иллюзию контроля, но скрывающих тот значимый шум миллиардов простых операций, находящихся на самом низком уровне абстракции и являющимися основой работы компьютера. Об этой процессуальности пишет Манович в уже упомянутой книге «Software Takes Command»: «В софтверной культуре мы не можем больше говорить о «документах», «работах», «сообщениях» или «звукозаписи» в терминах XX века. Вместо стабильных по форме документов, которые можно изучать с точки зрения структуры и содержания <…> сегодня мы сталкиваемся с динамичными “софтверными перформансами”, поскольку мы претерпеваем этот опыт в реальном времени. <…> Таким образом, хотя неизменяемые файлы могут задействоваться, в конечном счете опыт медиа, конструируемый софтвером, обычно не совпадает с восприятием одного отдельного файла, принадлежащего к определенному медиа». [Манович 2013: 28].

Чтобы избежать редукции своей практики к индивидуальному восприятию текста, на уровне репрезентации аналогичному восприятию с печатной страницы, но не отображающим те медиатехнические условия, в которых существует ее практика, mez совершает два действия. Первое из них заключается в способе распространения ее текстов – почтовой рассылке, которая включает тексты в сети циркуляции сообщений, встраивает их в аутопоэтические системы человеческой рецепции, создающие новые способы означивания и работы в медиатехническом контексте. Текст выходит из своей замкнутой системы и становится частью процесса коллективного интерпретирования и письма, распространяемого по децентрализованным протоколам обмена электронной почтой. Т.е. закрытость медиатехнических способов производства текста, характеризуемая возможностью манипулировать фрагментами текста любого размера и откатывать назад изменения, сочленяется с открытостью реципиента, которому приходит письмо с текстом, и которое он не может ни отнести к литературе (если это читатель, незнакомый с творчеством mez, а на раннем этапе ее творчества таких было большинство), ни отнести к категории уже существующих жанров писем, существующих в электронной переписке. Для читателя такой текст прерывает привычный поток информации, становится «слепым пятном» в привычном опыте взаимодействия в Интернете.

Второе действие, которое является продолжением первого – выпуск «итоговой» книги текстов под названием «Human Readable Messages» в 2011 году. Теперь тексты вынимаются из того окружения, в котором их существование и рецепция определялись заданными паттернами взаимодействия внутри различных интерфейсов. В бумажной форме они теряют некоторые возможности внешней читательской интеракции, связанные с редактированием текстов, удалением лишних знаков и «форматированием», подгонкой текста под себя, но при этом дают больше свободы интроэцированным способностям читателя, концентрируют внимание на аспектах рецепции, не выраженных при чтении этих текстов с экрана: например, на информационной избыточности текстов и отсутствии фильтров, способных эту избыточность компенсировать. Тексты предстают в минимально опосредованном виде. Кроме действия перемещения больших корпусов текстов из одного медиума в другой или изменению способов их распространения, mez совершает такого рода операции на уровне отдельных текстов или циклов.

Перейдем от методов, присущих языкам программирования и возможностям, предлагаемым различным программным обеспечением к особенностям текстов mez с точки зрения литературоведческого анализа. Сначала будет рассмотрен лингвистический аспект текстов mez, затем более общие стратегии встраивания литературы в современное медиатехническое окружение. Стоит отметить, что цитаты из текстов mez будут приведены в оригинале, поскольку перевод на русский с учетом всех необходимых для исследования аспектов невозможен.

Основными темами текстов mez можно назвать коммуникацию, контроль, и как они влияют на телесные и когнитивные аспекты человеческой жизни, каким образом формируют различные институции, контролирующие общество, на каком материально-техническом основании они разворачивают свои практики. Многие ее тексты повторяют по форме запись технического документа или кода программы.

На уровне морфологии тексты имитируют древовидную структуру кода программы – скобки, в которых заключаются различные морфемы, собираются в сложносоставные слова, которые можно читать различными способами, опуская те или иные морфемы: «ur[*||*the]myth.of.half.glass.finished.bio.mi[r rored]mickry.in.philop[astry/ like]hilosophistry.is.so[aked N wine-dangerous]old.s[hoal]kool-». [Бриз 2011: 88] Важно рассматривать аффиксацию в этих текстах не как средство выражения, увеличивающее возможности высказывания, но как усложнение процесса линейного чтения, интеоризация структур построения «высказывания» в языке программирования, опыта микротемпоральности исполнения программы, находящегося за пределами человеческого восприятия. Также присутствуют различные сокращения, характерные для переписки в чатах или по электронной почте: замена букв цифрами, «n» вместо «and»: «draw threads 2wards analytical reasoning rather then n.courage communication methods||fractures». [Там же: 75].

На уровне синтаксиса можно выделить такие особенности, как простые именные предложение, преобладание монопредикативных и полипредикативных распространенных простых предложений, практически полностью отсутствуют артикли и вспомогательные глаголы, преобладание настоящего времени глагола: «.watches as a bo[u]y rings his hands.in.cal[a]m[ity]ness.jerks. / .small.tr.a[r]cings.of.mother.loadings.echo.in.bo undless.sta.m.p[ed]es / .pledges.of.loss.+.aching.re:stricti[veness]on». [Там же: 110].

Можно выделить указание на временной характер медиа – статичен он только для человеческого взгляда, но по своей сути он динамичен – потоки информации постоянно обновляют экранное изображение (не только когда мы, например, перелистываем страницы, но и статичное изображение также постоянно обновляется, хотя этот процесс невидим для человека).

Чтобы подробней рассмотреть то, как mez концептуализирует время в своих текстах, обратимся к исследованиям нарратива с помощью теории систем, проведенной Брюсом Кларком. Кроме различных видов нарративного времени, описанных структуралистами, он выделяет время чтения, которое затратит наблюдающая система (читатель) на рецепцию текста. С одной стороны, это время процесса, который не повернуть вспять, но с другой ­– при чтении всегда можно перечитать предыдущий отрывок, вернуться к первоначальному месту и продолжить чтение. Таким же образом построены многие нарративные тексты, где повествование перемещается ретроспективно или проспективно по отношению к определенной сюжетной точке. Таким образом, текстовая анахрония гомологична обратимости структуры (в терминах системной теории), а фабульное время гомологично необратимому времени процесса. В текстах mez существует только фабула – они существуют только здесь и сейчас на мониторе и здесь и сейчас в своей технической реализуемости. Несмотря на большие возможности по созданию и форматированию текста в текстовом редакторе, эта многофункциональность не создает предпосылок для процессуальности, необратимости: любое изменение можно откатить, все действия уже предзаданны и ждут своей актуализации. В своих текстах mez пытается противостоять этой обратимости указанием на нее, репрезентированной в хаотичном смешении знаков. Таким образом, в своей практике она отказывается от скрытой обратимости, вписанной в софт для создания текста.

Далее рассмотрим важную проблему коммуникации, пронизывающую все работы mez. Вместо оппозиции отсутствия/наличия коммуникации, мы будем использовать иную оппозицию коммуникации/экскоммуникации, предложенную медиатеоретиками Ю. Такером, А. Гэллоуэем и М. Уорком. Экскоммуникация – это измерение коммуникации, единственным парадоксальным сообщением которой является отсутствие сообщения. В текстах mez присутствует измерение экскоммуникации, но оно скорее направленно не на сообщение того, что привычное линейное чтение этих текстов больше невозможно (это является следствием общей стратегии авторки), а невозможность привычной пользователю компьютера ответной реакции на его действия. Эта невозможность интериорезируется особым образом, через постоянное напряжение внимание в попытках проинтерпретировать считываемый текст, которое чаще всего заканчивается неудачей.

На уровне семантики часто присутствуют слова, связанные с телесным и техническим, присутствует военная лексика. Многие авторы сходятся на том, что тексты mez семантизируют формализм компьютерных языков, придавая множественность значений однозначным элементам языков программирования.

На уровне фонем также происходит интересная работа с языком – но не как в поэзии иных эпох и традиций, где актуализируются акустические возможности языка (от разложения на мельчайшие элементы в поэзии футуризма и леттризма до звукописи и ассонансов, существующих в других поэтических традициях). В текстах mez скорее обыгрывается разрыв между кодированием отдельного знака и его репрезентацией на экране. Чтобы рассмотреть эту тему подробней, используем противопоставление визуального и письменного, введенного немецким ученым Виллемом Флюссером. Он пишет о взаимодействии кода (или алгоритма) и письма: на примере научного текста он показывает, что есть различие при восприятии цифр и формул (отображающих идеи, воспринимаемые визуально) и букв (репрезентирующих звуки, воспринимаемые аудиально). В современном медиатехническом окружении одномерное буквенное письмо, прочитываемое единожды, сталкивается с двухмерным, которое воспринимается дважды, поскольку глаза воспринимают формулы как изображения. Сам текст становится изображением. Для компьютера визуальная информация не важна – он проверяет соответствия закодированного, кода и его значения – т.е. для него множественность значений существует в дискретном и последовательном виде, причем большая часть этой множественности различных кодировок недоступна пользователю. Экран, парадоксальным образом, статичен в своей постоянной обновляемости. Пользователь же, с одной стороны, не разбирает визуальные отличия английской и русской «а» на экране, но с другой, он идет дальше простой кодировки, множество значений для него существуют скорее в горизонтальной плоскости, а не в виде иерархий, как в коде, и поэтому переприсваивание значений происходит легче и в процессе чтения. Mez размывает старые паттерны восприятия чтения, вписывая в них машинные паттерны распознавания однозначной кодировки – многозначность создается при помощи симуляции процесса перемещения между различными кодировками, которые вызываются с помощью неконвенционального использования пунктуационных знаков.

На уровне общей стратегии можно выделить фокус на работе систем и того, как они конструируют образы, которыми пользователь может манипулировать в процессе чтения или более прямого взаимодействия с электронной литературой. Важно отметить, что сама авторка описывает свою практику в терминах, похожих на описание работы систем в одноименной теории: «Mez утверждает, что в ее [практику] следует мыслить в терминах “постепенного процесса фильтрации”». [Рэйли URL].

Подводя итоги, можно сказать, что ее работа с языком структурно относится также, как и описываемый Л. Мановичем процесс работы фильтра «Волна» в графическом редакторе «Photoshop». Как и фильтр с названием «Волна», являющийся компьютерным алгоритмом, использующим математические формулы для генерации синусоидальных кривых, создающим не только визуально похожие на волну паттерны, но и практически бесконечный набор непохожих на волны визуальных паттернов, так и mez использует возможности текстового редактора для градуального смещения отношений между возможностями интерфейса и стратегиями означивания. Ее тексты мобилизуют читательское усилие, но направленное не на более внимательную работу с текстом, прослеживанием всех возможных путей его прочтения – такого рода интерпретация является экстериоризацией известных еще постструктуралистам свойств любого текста, «сотканного» из множества кодов, – а на формирование навыка гиперчтения, в котором разные стратегии взаимодействия с литературой и компьютерными системами смешиваются. Каким образом это происходит, будет рассказано в следующем разделе.

**2.2.5 Амплификация и «остранение»**

Для того, чтобы пересмотреть взаимоотношения между терминами амплификации и «остранения», нужно вернуться к понятиям информации и шума. При работе с компьютером пользователь манипулирует определенным количеством информации, к которым, в идеале, не примешивается шум. Например, при работе с текстовым редактором он знает, что после нажатия на клавишу в поле ввода появится печатный знак, какие пункты меню отвечают за те или иные функции, и т.д. Он находится в петле отрицательной обратной связи, где информация на входе обрабатывается заданным образом, на выходе получается ожидаемый результат, который можно опять редактировать, и т.д. В такого рода коммуникации шум минимален, и выполнение операций происходит наиболее эффективно. С другой стороны, существует литературное письмо, которое переполнено «шумом», завершение интерпретации все время откладывается и дополняется новыми деталями, перестраивающими предыдущую картину. Такую точку зрения можно экстраполировать на положение литературы в современном медиатехническом окружении в целом, как это делает Уильям Поулсон, автор книги «Шум культуры: литература в мире информации». Литература, в его представлении, находится «на задворках» современной культуры, центрированной вокруг производства и циркуляции однозначно интерпретируемых, ясных сообщений. Но мы подойдем к этой проблеме с другой стороны. Нашей задачей является усложнением привычных оппозиций, где литература противопоставляется современному медиатехническому окружению с помощью добавления «шума» в свои «сообщения», или где литература понимается как разрыв коммуникации. Как медиахудожники и авторы работ э. литературы используют технологии не по назначению или создают свой, работающий на других основаниях софтвер, не связанный строгими причинно-следственными связями (например, проект WebStalker от группы I/O/D, работы коллектива Jodi и другие примеры нетарта и медиа-арта), так и литература может перенимать некоторые черты, характерные для особой организации опыта, существующей в программном обеспечении. В отличие от большинства работ э. литературы, где существует эксплицитное деление на методы работы (алгоритмы) и результат, и где существует неопределенность на уровне того, какой будет следующая итерация работы, mez не разделяет методы работы и материал, но на более высоком уровне абстракции (на уровне графического интерфейса) воспроизводит работу самого низкого уровня абстракции, машинного кода, где существуют простейшие арифметические операции с числовыми значениями. У нее это воспроизводится в виде результата работы простейших операций с текстом, доступных в текстовом редакторе: печать, выделить, копировать, вставить и некоторые другие. Читатель сталкивается не со сложной работой множества алгоритмов и протоколов, скрытых за графическим пользовательским интерфейсом и репрезентированном в виде хаотичной смеси различных символов, но с самими методами, формирующими этот хаос. Инструменты, созданные для того, чтобы работа в текстовом редакторе была быстрее, удобней и эффективней, при неконвенциональном использовании делают результат их работы нечитаемым. Можно сказать, что в тексте должны считываться не значения, присущие тому или иному языку, и произведением является не сам текст, но та последовательность простейших операций, участвовавших в создании текста. Их и нужно считывать при рецепции ее произведений, связывая с тем текстовым материалом, которым они образуют. От вопроса рецепции этих текстов следует перейти к проблеме исполнения, заявленной в начале второй главы и рассмотреть то, как это понятие трансформируется при введении оппозиции между исполнением (execution) и выполнением (performance), а также такими понятиями медиатеории, как эксплойт и текстовый вирус.

**2.3 Исполнение и выполнение**

Прежде чем обратиться к понятию исполнения, следует рассмотреть то, как функционируют языковые практики в современной медиатехнической среде. Александр Гэллоуэй в книге «Эффект интерфейса» проводит различие между тем, как натуральные языки *экстенсивно* работают с правилами определенных символических систем, т.е. в развертывании высказывания они работают с помощью присоединения различных дискурсов. В отличие от натуральных языков, в языках программирования правила символических систем воспроизводятся на практике или нарушаются *интенсивно*, т.е. внутри и с помощью этих правил, без работы на стыке нескольких дискурсов. На примере эксплойта он показывает, что вредоносный код может использовать существующие уязвимости программы в целях приостановки работы этой программы. Например, при DDoS-атаке множество компьютеров выполняют слишком большое количество обращений к какому-либо серверу, что приостанавливает его работу. Атака совершается иным использованием сети, которая в этом случае, как пишет Гэллоуэй, работает слишком хорошо и не оставляет ни один запрос без внимания, поэтому сервер и перестает нормально функционировать. Интенсивность по большей части завязана на формализации и числовой репрезентации, переводу любой информации в двоичные значения. На первый взгляд такого рода работа с языком противоречит привычным стратегиям означивания, всегда диалогичным, открытым к взаимодействию с другими текстами и работами иных медиа. Но в случае электронной литературы, существующей неразрывно от медиатехнического окружения и при этом задействующей стратегии означивания, присущие обычной литературе, нельзя сразу отбросить это понятие и разводить практики вычисления и письма. Чтобы глубже разобраться в проблеме, мы рассмотрим ее через оппозицию выполнения (performance) и исполнения (execution).

Стоит начать с этимологии этих слов. Слово «perform» произошло от старофранцузского слова «parfornir», означающего «делать, осуществлять, завершать». Первые примеры употребления этого слова относятся к XIV веку, в XVII оно начинает употребляться в значении «играть на сцене», «петь» или «играть на музыкальном инструменте». В данный момент словом перформанс также обозначают определенные художественные практики, ключевой частью которых является действие художника в определенном пространстве и времени. Слово execution происходит от французского выражения XII века «l’executeur du testament», означающего исполнителя некой воли. Контекстом употребления этого слова было исполнение судебных предписаний и других бюрократических процедур. Интересно, что это перекликается со сравнением внутренней работы компьютера с бюрократическими службами. Слово исполнение «всегда относится к настоящему [моменту], актуализации, присутствию, уже всегда закончившемуся». [Critical Software Thing 2018:142]. Таким образом, различие между выполнением и исполнением в том, что первое имеет более широкий смысл и может употребляться не только в контексте искусства и театра, при этом обычно применяется к длящимся по времени событиям или действиям, а второе применяется к действию, происходящему мгновенно.

Если применить эти термины к предыдущему разбору различных методов, используемых mez при создании текстов, можно соотнести выполнение (performance) как относящееся к методам производства значения, описанном в предыдущей части работы, а исполнением – с системами амплификации. Чтобы теперь рассмотреть, какого рода исполнение происходит в текстах, мы возьмем понятие текстового вируса, а затем мы вернемся к понятиям экстериоризации и интериоризации, важных для теории медиа в целом.

Понятие текстового вируса ввел Марко Дезерис. Оно означает введение в заблуждение пользователя текстовой инструкцией, исполнение которой приведет к прекращению работы компьютера. Такого рода вирусы были распространены в 90-ых годах XX века. Примером может служить сообщение «на ваш компьютер загружен вирус, чтобы избавиться от него, удалите папку system32 на диске C вашего компьютера». На самом деле удаление этой важной системной папки приведет к поломке компьютера. Согласно Дезерису, такого рода ошибки во взаимодействии происходят по причине тесного сосуществования кода и естественного языка. Пользователь корректно считывает сообщения на естественном языке, но не знает, как действие согласно этой инструкции повлияет на исполняемый код. Текстовый вирус, таким образом, приостанавливает процесс полисемии, на уровне естественного языка являя собой однозначно интерпретируемую команду, но на уровне исполнения являющийся чем-то противоположным. Он эксплуатирует доверчивость пользователя, привыкшего незамедлительно видеть *ожидаемый* результат своих действий. Исполнение текстового вируса дает результат, но не тот, который ожидал пользователь.

Если применить это понятие к текстам mez, то можно сказать, что ее тексты служат своего рода текстовыми вирусами. Надо отметить, что такого рода понимание отличается от похожей трактовки Флориана Крамера, предложенной в статье «mez, \_Виро.Логическое Обусловли] [вание] [1.1\_ Анализ Текста». Крамер использует слово «заражение» в метафорическом смысле – для него гибрид языков программирования и английского языка является метафорой заражения одного языка другим, проникновение в «тело» английского языка «вируса» формализованных языков программирования, изменяющих кардинально процессы рецепции. Понятие текстового вируса помогает уточнить то, какого рода исполнение происходит в текстах mez и не метафоризировать внутритекстовые отношения. Текстовые вирусы, которыми являются тексты mez, работают по-другому: они создают «короткое замыкание», постоянно воспроизводящее разложение текста на простые операции чтения с экрана/записи внутри памяти компьютера, которое для читателя проявляется как микротемпоральность, *останавливающая* процесс чтения, герменевтическая петля обратной связи, в которой процесс гиперчтения, как его описывает Хэйлс, дается как таковой, без наполняющего его текстового содержания, готового к последующей когнитивной обработке.

Здесь появляется проблема памяти и того, как она соотносится с подобным чтением. В статье «Как мы читаем» Кэтрин Хэйлс описывает возможные причины того, почему понимание текста с веб страницы хуже, чем если бы он читался с листа бумаги. Она связывает это с изменившимися отношениями между краткосрочной и долгосрочной памятью. При обычном чтении первичное восприятие текста сохраняется в краткосрочной памяти, и для дальнейшего осмысления передается в долговременную память. Этот линейный процесс, при котором нагрузка на кратковременную память минимальна, прерывается при чтении текста с веб страницы, поскольку когнитивная нагрузка на кратковременную память увеличивается посторонними действиями: скроллингом, переходом по гиперссылкам, и таким образом уменьшается объем запоминаемой информации. При обычном чтении нагрузка на краткосрочную память минимальна, потому что «движения глаз более рутинны и принимают меньше решений о том, как читать материал и в каком порядке». [Хэйлс URL]. Парадоксально, но чем больше интеракция, выбор, тем меньше осмысления. т.е. нужна некая однородность (для взгляда, нейтральность), чтобы внимание и понимание были на высоком уровне. Тексты mez – это своего рода выравнивание, интенсивное сведение множества возможных контекстов к нескольким методам, доступным пользователю в интерфейсе, новый способ интериоризации исполнения текста, возможность рекомбинации уже существующих инструментов, направленная не вовне, к созданию иных способов работы с технологиями, при этом с сохранением дистанции от исполняемого, исчисляемого, с настаиванием на том, что всегда существует нечто невычисляемое, выходящее за пределы формализации, но внутрь технологий, понимание текста как закодированного, формализованного и не имеющего внешнее измерение. Такой подход может показаться редукционистским, не учитывающим множество других отношений текста и медиатехнического окружения. Но поскольку сами одной из главных черт современных компьютерных технологий является числовое представление любой информации, то mez идет дальше в своей редукции производства и рецепции текстов, доводя до абсурда фразу Вэнди Чан из книги «Запрограммированное зрение» «чтение – это письмо в другом месте». Рецепция текстов приравнивается к «письму», т.е. постоянному обновлению данных, считывающих активность пользователя, к процессам, происходящим обычно *вне* читательских когнитивных, языковых и социальных систем. Рецепцию ее текстов составляет не только «гиперчтение» или машинное чтение, сканирующее текст, разбирающее его на составные синтаксические, лексические, морфологические и фонетические части, но и требующее видеть результаты применения простых инструментов работы с текстом, существующих в текстовых редакторах. Она, таким образом, дает не только почувствовать опыт воздействия информационного шума, превосходящего субъекта и вводящего в состояние дезориентации, но и интериоризирует герменевтическую петлю обратной связи, присущую нижнему уровню абстракции, машинному коду, который выражается в исполнении простого алгоритма чтения и приостановки чтения – противоречие между интериоризированным способом чтения, постоянно прерываемым и не укладывающимся в рамки микротемпоральностей, в которых существует машинный код, и является основным эффектом, исполнением, не сводимому к эксплицитной работе кода в программной среде компьютера.

**Заключение**

Исследования электронной литературы должны учитывать не только основные признаки литературного текста, его темы, стратегии означивания, возможные способы рецепции, но и медиатехническое окружение, которое включает в себя не только конкретные инструменты, протоколы и интерфейсы, но и те абстракции, которые лежат в их основе.

Спустя несколько десятилетий после появления текстов mez, когда стремительное развитие технологий опережает их теоретическое осмысление, нужно не только теоретизировать настоящие технологии и результаты их применения в искусстве и литературе, но и возвращаться к уже будто бы исследованным, вписанным в определенную историческую рамку, будто бы не могущим нам сказать нечто интересное и важно о сегодняшней ситуации. Одним из примеров такой практики является mezangelle – своего рода код, в котором сосуществует несколько разнородных методов производства значения и действия, созданный австралийской писательницей Мэри Энн Бриз, более известная под псевдонимом mez. Ее тексты несут на себе отчетливые приметы времени, способы общения и используемые программы, сегодня признанные устаревшими или изменившими свою функцию. Современная литературная теория, искусствоведение и медиатеория дали ее работам однозначное описание, с ее текстами соотносят определенные термины, такие как исполнение, код, пиджин, code work, и некоторые другие. Особенно выделяются из них два: исполнимость и гибрид, которые вызвали полемику среди сообщества теоретиков и практиков э. литературы. Исполняемость/неисполняимость трактовались как строгая оппозиция между рабочим кодом и простым текстом на поверхности интерфейса, или как взаимодополняемость этих модусов существования, в которых неисполняемый текст включается в работу кода как неисполняемый фрагмент, вызывающий определенную реакцию у системы. Согласно этим трактовкам тексты mez или не являются кодом, или определяются как потенциальный код, включающий в себя потенциальную возможность быть исполнимым. Понятие гибрида или языка-пиджина означает смешение содержаний естественного языка и различных языков программирования, в зависимости от интерпретации формирующих новый язык или составляющих коллаж.

В данном исследовании была проделана работа по пересмотру оппозиции понятий исполнимости/неисполнимости и понятия гибрида. Вместо них были описаны оппозиция performance/execution и понятие медиа-гибрида. Первое из них описывает противопоставление экстенсивной работы между несколькими разнородными дискурсами, оставляющими следы и имеющие характер и необратимость процесса, и интенсивной работой внутри возможностей одного языка, формализующие другие дискурсы и сводящие их в один формальный язык, составляющий систему, в которой практически любое изменение состояний обратимо. Если применить эту оппозицию к текстам mez, то можно установить, что в них преобладает интенсивная работа с языком, создающая новые типы интериоризации информации, ее разложения на простые операции чтения/письма, которые делают видимым для компьютера действия, интериоризируемые в когнитивную систему читателя. Понятия гибрида было заменено на понятие медиа-гибрида, взятое из работы Льва Мановича «Софтвер берет на себя командование». В текстах mez перемешиваются не содержания языков, а те методы, которыми они оперируют при формулировании высказываний или создании значений, получается нечто третье, обладающее уникальными свойствами и способами работы на границе этих двух языков, а не просто чередование языковых систем на различных, от фонетики до синтаксиса, уровнях языка, принуждающих читателя переключать внимание со способа означивания, присущего английскому языку на способ организации знаков, существующий в языках программирования. В текстах mez разрабатываются новые способы взаимодействия между амплификацией пользователя, присущей компьютерным системам, и тем, что в первом приближении можно назвать постоянным «нарушением ожиданий», несовпадением между установками читателя и читаемым текстом.

# **Список использованной литературы:**

1. Breeze M. Human Readable Messages. – Traumawein, 2011. – 327 p.
2. Брилов Д., Брукшир Дж. Гленн. Компьютерные науки. Базовый курс. – СПб.: ООО «Диалектика», 2019. – 992 с.
3. Глик Дж. Информация. История. Теория. Поток. – М.: АСТ: CORPUS, 2013. – 576 с.
4. Киттлер Ф. Оптические медиа. – М.: Изд-во «Логос», 2009. – 272 с.
5. Крамер Ф. Концепты, системы записи, программное обеспечение, искусство [Электронный ресурс]. – URL: http://web.archive.org/web/20030407063143/http://www.macros-center.ru /read\_me/teb2.htm. – (Дата обращения: 14.07.2020).
6. Маклюэн Г. М. Понимание медиа: Внешние расширения человека. М.: Жуковский: «КАНОН-пресс-Ц», «Кучково поле», 2003. – 464 с.
7. Ман, П. де. Аллегории чтения: Фигуральный язык Руссо, Ницше, Рильке и Пруста. – Екатеринбург: Изд-во Урал. Ун-та, 1999. ­– 368 с.
8. Манович Л. Язык новых медиа. – М.: АД МАРГИНЕМ ПРЕСС, 2018. – 400 с.
9. Мечковская Н. Б. Семиотика. Язык. Природа. Культура. – М.: Издательский центр «Академия», 2007. – 432 с.
10. Орехов Б. В. Стихи в программном коде [Электронный ресурс]. – URL: http://nevmenandr.net/personalia/21.pdf. – (Дата обращения: 16.09.2020).
11. Пол К. Цифровое искусство. – М.: АД МАРГИНЕМ ПРЕСС, 2017. – 272 с.
12. Таненбаум Э.,Остин Т. Архитектура компьютера. ­– СПб.: Питер, 2013. – 816 с.
13. Хуэй Ю. Рекурсивность и контингентность. – М.: V-A-C Press, 2020. – 400 с.
14. Эскелинен М. Игровая ситуация [Электронный ресурс]. – URL: https://gamestudies.ru/translations/gamingsituation. – (Дата обращения: 20.05.2020).
15. Berry D. M. The Philosophy of Software: Code and Mediation in the Digital Age. – Palgrave Macmillan, 2011. – 200 p.
16. Breeze M. \_Non Compos Mentis: Zen-Tripping the Non-Conference Circuitry\_ [Электронный ресурс]. URL: <http://web.archive.org/web/20070624080735/http://www.fineartforum.org/Backissues/Vol_15/faf_v15_n05/text/feature.html>. – (Дата обращения: 19.02.2020).
17. Cayley J. The Code Is Not the Text (Unless It Is the Text) // Cayley J. Grammalepsy: Essays on Digital Language Art. – Bloomsbury Academic, 2018. – P. 53-67.
18. Cayley J. Time Code Language // Cayley J. Grammalepsy: Essays on Digital Language Art. – Bloomsbury Academic, 2018. – P. 95-115.
19. Chun W. H. K. On «Sourcery», or Code as Fetish // Configurations. – 2008. – Vol. 16. – № 3. – P. 299-324.
20. Chun W. H. K. On Software, or the Persistence of Visual Knowledge [Электронный ресурс]. – URL: <https://monoskop.org/images/9/9b/Chun_Wendy_Hui_Kyung_2005_On_Software_or_the_Persistence_of_Visual_Knowledge.pdf>. – (Дата обращения: 22.06.2020).
21. Chun W. H. K. Programmed visions: software and memory. – Cambridge, Mass: MIT Press, 2011. – 239 p.
22. Clarke B. Communication // Critical Terms for Media Studies / eds. M. B. N. Hansen, W. J. T. Mitchell. – University of Chicago Press, 2010. – P. 131-145.
23. Clarke B. Information // Critical Terms for Media Studies / eds. M. B. N. Hansen, W. J. T. Mitchell. – University of Chicago Press, 2010. – P. 157-172.
24. Clarke B. Neocybernetics and narrative. – London: University of Minnesota Press, 2014. – 211 p.
25. Clarke B. Systems theory // Routledge companion to literature and science / eds. B. Clarke, M. Rossini. – New York: Routledge, 2011. – P. 214-226.
26. Cox G., McLean A. Speaking code: coding as aesthetic and political expression. – The MIT Press, 2013. – 149 p.
27. Cox G., Soon W. Aesthetic Programming: A Handbook of Software Studies. – Open Humanities Press, 2020. – 296 p.
28. Cramer F. Digital Code and Literary Text [Электронный ресурс]. – URL: http://beehive.temporalimage.com/content\_apps43/cramer/oop.html. – (Дата обращения: 18.12.2019).
29. Cramer F. mez, \_Viro.Logic Condition] [ing] [1.1. // Cramer F. Anti-media: ephemera on speculative arts. – NY: nai010 publishers, 2013. – P. 167-186.
30. Cramer F. The Foul Promises of ‘Interactivity’ and ‘Openness’. Rereading ‘Art, Power and Communication’. // Cramer F. Anti-media: ephemera on speculative arts. – NY: nai010 publishers, 2013. – P. 20-24.
31. Cramer F. Words Made Flesh: Code, Culture, Imagination. – Piet Zwart Institute, 2005. ­– 141 p.
32. Cramer F., Gabriel U. Software Art [Электронный ресурс]. – URL: https://www.netzliteratur.net/cramer/software\_art\_-\_transmediale.html. – (Дата обращения: 01.06.2020).
33. Critical Software Thing. Execution // Posthuman Glossary / eds. R. Braidotti, M. Hlavajova. – Bloomsbury Academic, 2018. – P. 141-145.
34. Curruthers A. Mez Breeze, Human Readable Messages (Mezangelle 2003-2011) // Southerly. – 2012. – Vol. 72. – № 3. – P. 31-36.
35. Deseriis M. Text Virus // Software studies: a lexicon / ed. M. Fuller. – Cambridge, Mass: MIT Press, 2008. – P. 250-256.
36. Drucker J. Diagrammatic Writing [Электронный ресурс]. – URL: <http://ubu-mirror.ch/media/text/vp/drucker_diagrammatic_writing_2013.pdf>. (Дата обращения: 12.07.2020).
37. Evans S. The Anti-Logos Weapon: Excesses of Meaning and Subjectivity in Mezangelle Poetry [Электронный ресурс]. – URL: <http://cordite.org.au/essays/anti-logos-weapon>. (Дата обращения: 12.11.2020).
38. Flores L. Third Generation Electronic Literature [Электронный ресурс]. – URL: <http://electronicbookreview.com/essay/third-generation-electronic-literature>. – (Дата обращения: 06.10.2020).
39. Florian Cramer: Program Code Poetry [Электронный ресурс]. – URL: https://www.netzliteratur.net/cramer/programm. – (Дата обращения: 18.12.2019).
40. Flusser V. Does Writing Have a Future? – University of Minnesota Press, 2011. – 208 p.
41. Fuller M. Behind the blip: essays on the culture of software. – Brooklyn, NY: Autonomedia, 2003. – 165 p.
42. Gabrys J. Afterword: Reverse Executions in the Internet of Things // Executing Practices / eds. H. Pritchard, E. Snodgrass, M. Tyzlik-Carver. – Open Humanities Press, 2018. – P. 295-303.
43. Galloway A. R. Networks // Critical Terms for Media Studies / eds. M. B. N. Hansen, W. J. T. Mitchell. – University of Chicago Press, 2010. – P. 280-297.
44. Galloway A. R. Protocol: how control exists after decentralization. – MIT Press, 2004. – 260 p.
45. Galloway A. R. The Interface Effect. – Polity, 2012. – 182 p.
46. Galloway A. R., Thacker E. The Exploit: A Theory of Networks. – University of Minnesota Press, 2007. – 256 p.
47. Galloway A. R., Thacker E., Wark M. Introduction: Execrable Media // Galloway A. R., Thacker E., Wark M. Excommunication: Three Inquiries in Media and Mediation. – University of Chicago Press, 2013. – P. 1-25.
48. Gauthier D. On Commands and Executions: Tyrants, Spectres and Vagabonds // Executing Practices / eds. H. Pritchard, E. Snodgrass, M. Tyzlik-Carver. – Open Humanities Press, 2018. – P. 69-85.
49. Goffey A. Algorithm // Software studies: a lexicon / ed. M. Fuller. – Cambridge, Mass: MIT Press, 2008. – P. 15-21.
50. Greene R. Web Work: A History of Internet Art [Электронный ресурс]. – URL: <https://www.artforum.com/print/200005/web-work-a-history-of-internet-art-465>. – (Дата обращения: 15.10.2020).
51. Hansen M. B. N. New philosophy for new media. – Cambridge, Mass: MIT Press, 2004. – 333 p.
52. Hansen M. B. N., Mitchell W. J. T. Introduction // Critical Terms for Media Studies / eds. M. B. N. Hansen, W. J. T. Mitchell. – University of Chicago Press, 2010. – P. vii-3.
53. Hayles N. K. Cognition Everywhere: The Rise of the Cognitive Nonconscious and the Costs of Consciousness // New Literary History. – 2014. – № 2 (45). – P. 199–220.
54. Hayles N. K. Electronic Literature: New Horizons for the Literary. – University of Notre Dame Press, 2008. – 223 p.
55. Hayles N. K. Electronic Literature: What is it? [Электронный ресурс]. – URL: <https://eliterature.org/pad/elp.html>. – (Дата обращения: 12.10.2020).
56. Hayles N. K. How We Read: Close, Hyper, Machine // ADE Bulletin. – 2010. – № 150. – P. 62-79.
57. Hayles N. K. Literary Texts as Cognitive Assemblages: The Case of Electronic Literature [Электронный ресурс]. – URL: https://electronicbookreview.com/essay/literary-texts-as-cognitive-assemblages-the-case-of-electronic-literature. – (Дата обращения: 30.08.2020).
58. Hayles N. K. Print is Flat, Code is Deep: The Importance of Media-Specific Analysis // Poetics Today. – 2004. – 25.1. – P. 67-90.
59. Hayles N. K. Virtual Bodies and Flickering Signifiers // October. – 1993. ­– Vol. 66. – P. 69-91.
60. Heylighen F., Joslyn C. Cybernetics and Second-Order Cybernetics [Электронный ресурс]. – URL: <https://www.researchgate.net/publication/2354032_Cybernetics_and_Second-Order_Cybernetics>. – (Дата обращения: 15.06.2020).
61. Hui Y. On the existence of digital objects. – Minneapolis: University of Minnesota Press, 2016. – 314 p.
62. Hui Y. Preface: The Time of Execution // Executing Practices / eds. H. Pritchard, E. Snodgrass, M. Tyzlik-Carver. – Open Humanities Press, 2018. – P. 25-35.
63. Kirschenbaum M. G. Hello worlds: Why humanities students should learn to program [Электронный ресурс]. – URL: <https://www.chronicle.com/article/hello-worlds>. – (Дата обращения: 07.11.2020).
64. Manovich L. Software Takes Command. – Bloomsbury Academic, 2013. – 376 p.
65. Marino M. C. Code as Ritualized Poetry: The Tactics of the Transborder Immigrant Tool [Электронный ресурс]. – URL: <http://www.digitalhumanities.org/dhq/vol/7/1/000157/000157.html>. – (Дата обращения: 06.07.2020).
66. Marino M. C. Critical Code Studies [Электронный ресурс]. – URL: <http://electronicbookreview.com/essay/critical-code-studies>. – (Дата обращения: 12.03.2020).
67. Marino M. C. Reading Culture Through Code [Электронный ресурс]. – URL: <https://www.researchgate.net/publication/337458554_Reading_Culture_through_Code>. – (Дата обращения: 20.09.2020).
68. Memmot T. Digital Rhetoric and Poetics: Signifying Strategies in Electronic Literature [Электронный ресурс]. – URL: <https://muep.mau.se/bitstream/handle/2043/12547/2043_12547%20Memmott%20muep.pdf?sequence=2&isAllowed=y>. – (Дата обращения: 30.05.2020).
69. Mezangelle: Mez Breeze [Электронный ресурс]. – URL: <https://anthology.rhizome.org/mez-breeze>. (Дата обращения: 17.02.2020).
70. Myers R. Human Readable Messages [Электронный ресурс]. – URL: <https://www.furtherfield.org/human-readable-messages>. – (Дата обращения: 27.09.2020).
71. Quaranta D. Beyond New Media art. – Brescia: LINK Editions, 2013. – 281 p.
72. Raley R. Interferences: [Net.Writing] and the Practice of Codework [Электронный ресурс]. –URL: https://electronicbookreview.com/essay/interferences-net-writing-and-the-practice-of-codework. – (Дата обращения: 08.06.2020).
73. Reep J. "Re: The Fact That I Am Fiction": Mary-Anne Breeze, Her Avatars, and the Transformation of Identity [Электронный ресурс]. – URL: <https://quod.lib.umich.edu/p/postid/pid9999.0004.107/--re-the-fact-that-i-am-fiction-mary-anne-breeze-her-avatars?rgn=main;view=fulltext>. – (Дата обращения: 12.11.2020).
74. Rettberg S. Electronic Literature. – Polity, 2019. – 240 p.
75. Sack W. Memory // Software studies: a lexicon / ed. M. Fuller. – Cambridge, Mass: MIT Press, 2008. – P. 184-193.
76. Shulgin A. Art, Power and Communication [Электронный ресурс]. – URL: <https://v2.nl/archive/articles/art-power-and-communication>. – (Дата обращения: 17.02.2020).
77. Snodgrass E. What is Executing Here? // Executing Practices / eds. H. Pritchard, E. Snodgrass, M. Tyzlik-Carver. – Open Humanities Press, 2018. – P. 237-261.
78. Sondheim A. Codework [Электронный ресурс]. – URL: <https://www.litline.org/ABR/issues/Volume22/Issue6/sondheim.pdf>. – (Дата обращения: 12.03.2020).
79. Stefans B. K. Against Desire: Excess, Disgust and the Sign in Electronic Literature [Электронный ресурс]. – URL: <http://electronicbookreview.com/essay/against-desire-excess-disgust-and-the-sign-in-electronic-literature>. – (Дата обращения: 30.04.2020).
80. Stiegler B. Memory // Critical Terms for Media Studies / eds. M. B. N. Hansen, W. J. T. Mitchell. – University of Chicago Press, 2010. – P. 64-88.
81. Szilak I. Towards Minor Literary Forms: Digital Literature and the Art of Failure [Электронный ресурс]. – URL: http://electronicbookreview.com/essay/towards-minor-literary-forms-digital-literature-and-the-art-of-failure. – (Дата обращения: 09.01.2020).
82. Tabbi J. E-literature // Routledge companion to literature and science / eds. B. Clarke, M. Rossini. – New York: Routledge, 2011. – P. 214-226.
83. Tenen D. Plain text: the poetics of computation. – Stanford University Press, 2017. – 268 p.
84. Wildman P. Deconstructing code that works [Электронный ресурс]. – URL: <http://unsworks.unsw.edu.au/fapi/datastream/unsworks:49712/SOURCE02?view=true>. – (Дата обращения: 18.04.2020).