

---

# Cyber X. Описание жизненного цикла и сопровождения продукта

---

# Содержание

<b>Сyber X. Описание жизненного цикла и сопровождения продукта</b> .....	1
<b>Введение</b> .....	1
<b>Описание системы контроля версий</b> .....	1
Общее описание системы контроля версий .....	1
Модель работы .....	1
Доступ к репозиторию .....	2
Стадии разработки ПО Cyber X .....	2
Сборка программного кода ПО .....	3
Технология создания дистрибутива ПО .....	4
Элементы дистрибутива ПО .....	4
<b>Система отслеживания ошибок и порядок обновления ПО</b> .....	5
Общее описание и назначение .....	5
Техническая архитектура .....	5
Общий алгоритм обновление ПО .....	6
Порядок установки критических обновлений .....	7
Характеристики репозитория изделия .....	7
<b>Способы распространения изделия</b> .....	8
<b>Описание штатной структуры проекта</b> .....	8

# Cyber X. Описание жизненного цикла и сопровождения продукта

## Введение

Программное обеспечение (далее ПО) Cyber X это удобно и гибко настраиваемое решение для построения комплексных систем технической безопасности, включающих такие подсистемы как видеонаблюдение, охранная сигнализация, система управления и контроля доступом.

В настоящем документе представлены описания системы управления жизненным циклом Cyber X, которая используется в ООО «Норд-М» при разработке данного ПО и методы уникальной идентификации элементов выпускаемого ПО.

---

## Описание системы контроля версий

### Общее описание системы контроля версий

При разработке ПО для обеспечения контроля версий применяется программный продукт Mercurial (также известный как Hg), который является свободно распространяемой по лицензиям GNU/GPL кроссплатформенной распределенной системой управления версиями, официально выпущенной в 2005 году.

В ООО «Норд-М» при разработке ПО используется система Mercurial версии 4.2.3.

---

### Модель работы

Mercurial является распределённой (децентрализованной) системой контроля версий. Это означает, что рабочий процесс, как правило, выглядит следующим образом:

1. На личном компьютере создаётся новый репозиторий (путём клонирования существующего репозитория, создания нового и т. п.);
2. В рабочем каталоге данного репозитория изменяются/добавляются/удаляются файлы;
3. Выполняется фиксация (commit) изменений в данный репозиторий (то есть в локальный репозиторий на личном компьютере);
4. Шаги 2 и 3 повторяются столько раз, сколько необходимо;
5. Забираются (pull) чужие наборы изменений;
6. Производится слияние изменений (merge);
7. Отдаются (push) собственные.

То есть вся повседневная работа происходит в локальном репозитории, а когда возникает необходимость, производится отправка результатов своей работы в один или несколько

---

других репозиториях. При этом центральный репозиторий служит как референсный репозиторий, через который проходят все изменения из локальных репозиториях, что гарантирует внесение всех изменений от всех разработчиков, их корректное слияние.

Сборка ПО для тестирования (начиная с бета-версии) и дистрибуции осуществляется исключительно из исходных кодов центрального репозитория.

---

## Доступ к репозиторию

Mercurial предоставляет следующие способы обмена данными между репозиториями:

- через HTTPS;
- вручную (взаимодействие непосредственно с файловой системой) при помощи упакованных наборов изменений;
- через стандартный ввод-вывод (в том числе через средства удаленного CLI, например SSH).

Для доступа к одному и тому же репозиторию могут одновременно использоваться разные способы.

В ООО «Норд-М» доступ к удаленным (не локальным) репозиториям системы Mercurial обеспечивается только внутри контролируемой зоны офиса ООО «Норд-М».

---

## Стадии разработки ПО Cyber X

В разработке программного обеспечения, стадии разработки программного обеспечения используются для описания степени готовности программного продукта к его промышленной эксплуатации. Также стадия разработки может отражать количество реализованных функций, запланированных для определённой версии программы.

Стадии либо могут быть официально объявлены и регламентируются разработчиками, либо иногда этот термин используется неофициально для описания состояния продукта.

### Пре-альфа

Начальная стадия разработки — период времени со старта разработки до выхода стадии Альфа (или до любой другой, если стадия Альфа была пропущена). Также так называются модули ПО, не вышедшие еще в стадию альфа или бета, но прошедшие стадию разработки, для первичной оценки функциональных возможностей в действии.

В отличие от альфа и бета версий пре-альфа может включать в себя не весь спектр функциональных возможностей программы. В этом случае подразумеваются все действия, выполняемые во время проектирования и разработки программы вплоть до тестирования. К таким действиям относятся - разработка дизайна, анализ требований, собственно разработка приложения, а также отладка отдельных модулей.

## Альфа

Внутреннее тестирование — стадия начала тестирования программы в целом специалистами-тестерами, обычно не разработчиками программного продукта, но, как правило, внутри организации-разработчика. Также это может быть стадия добавления новых функциональных возможностей. Программы на данной стадии могут применяться только для ознакомления с будущими возможностями.

## Бета

Публичное тестирование - Стадия активного бета-тестирования и отладки программы, прошедшей альфа-тестирование (если таковое было). Программы этого уровня могут быть использованы другими разработчиками программного обеспечения для испытания совместимости. Тем не менее, программы этого этапа могут содержать достаточно большое количество ошибок.

Поскольку бета-продукт не является финальной версией, и публичное тестирование производится на страх и риск пользователя, производитель не несёт никакой ответственности за ущерб, причинённый в результате использования бета-версии.

## Релиз-кандидат

Релиз-кандидат или RC (англ. release candidate), Пре-релиз или Pre - стадия-кандидат на то, чтобы стать стабильной. Программы этой стадии прошли комплексное тестирование, благодаря чему были исправлены все найденные критические ошибки. Но в то же время существует вероятность выявления ещё некоторого числа ошибок, не замеченных при тестировании.

## Релиз

Релиз или RTM (англ. release to manufacturing промышленное издание) — издание продукта, готового к тиражированию. Это стабильная версия программы, прошедшая все предыдущие стадии, в которых исправлены все найденные ошибки, но существует вероятность появления новых, ранее не обнаруженных ошибок. RTM предшествует общей доступности, когда продукт выпущен для общественности.

---

## Сборка программного кода ПО

Сборка программного кода ПО производится на отдельной виртуальной ОС, размещенной на специальном выделенном сервере в контролируемой зоне производственного объекта ООО «Норд-М».

Доступ к виртуальной ОС имеют только определенные лица администраторов отделения разработки ОС, в круг обязанностей, которых входит работа со сборкой программного кода или

иных задач по контролю за работой над разработкой кода ПО. Доступ осуществляется на основе логина и пароля пользователя.

Решение о необходимости сборки программного кода принимает руководитель отдела разработки после соответствующей проверки и тестирования всех компонентов ПО.

Каждая сборка ПО имеет уникальный идентификатор, состоящий из назначенного номера версии ПО, даты и времени выполнения сборки (BDT - Build Data & Time).

---

## Технология создания дистрибутива ПО

При создании ПО используются следующие средства технологического оснащения:

- персональный компьютер с архитектурой процессора x86\_64;
- операционная система сервера сборки;
- cmake-3.10.1 - система управления сборкой;
- install-4.1.38 - средство построения и упаковки дистрибутива;

Методы эксплуатации средств технологического оснащения при создании ПО:

- создаётся или модифицируется проект ранее созданного дистрибутива в программе install (для каждой поддерживаемой ОС);
- из полученного проекта создаются (собираются) дистрибутивы для каждой из поддерживаемой ПО ОС;
- полученные дистрибутивы устанавливаются на чистые тестовые ОС, исправляются ошибки инсталляции;
- установленные дистрибутивы настраиваются на использование тестового оборудования, вводятся тестовые настройки и проверяется функционал и работоспособность самого ПО;
- после успешного тестирования процесса установки и успешного тестирования функционирования установленного ПО созданные дистрибутивы фиксируются как рабочие (с учетом стадии производства).

Правила эксплуатации средств технологического оснащения при создании ПО:

- необходимо соблюдение правил электро-технической безопасности при использовании сборочного персонального компьютера и тестового оборудования;

---

## Элементы дистрибутива ПО

Исходными элементами дистрибутива ПО являются:

- исходный код ПО;
- медиа-файлы оформления (шрифты, иконки, изображения);
- проектная документация;
- тестовая документация;
- руководства пользователя;
- референсная база данных;

- документация дистрибутива.

Собранный для распространения дистрибутив ПО включает:

- бинарные файлы модулей и библиотек;
- медиа-файлы оформления (шрифты, иконки, изображения);
- руководства пользователя;
- референсную базу данных;
- скрипт установки (разворачивания) модулей ПО, их библиотек, первичной настройки.

---

## Система отслеживания ошибок и порядок обновления ПО

### Общее описание и назначение

При разработке ПО использована система отслеживания ошибок (СОО) на базе свободно распространяемого ПО OTRS версии 6.

СОО предназначена для отслеживания одного или нескольких проектов, в которых циркулируют информационные элементы, которыми могут являться фиксация обращений клиентов, обращения сотрудников отдела тестирования и технической поддержки, поручения руководителей, данные о сбоях, обнаруженных в продукте при контроле качества и т.д.

Цикл обработки информационных элементов может проходить при участии различных программных комплексов в организации и персонала. СОО развернута на собственном сервере ООО «Норд-М» и доступ к ней возможен только в пределах контролируемой зоны производственного объекта ООО «Норд-М».

Ключевой сущностью в СОО являются информационные элементы, факты, которые определяют следующие группы:

- формализованные данные (поля, подструктуры, классификация);
- неформализованные данные (обсуждения, связи и т.п.);
- формальный жизненный цикл.

### Техническая архитектура

СОО реализована с применением двухзвенной архитектуры (AS с WEB GUI + DB). Решение легко интегрируется в существующую IT-структуру ООО «Норд-М» благодаря серверу СОО, имеющему открытые интерфейсы на основе открытых стандартов коммуникации и обмена информацией.

СОО может быть использована как:

- система управления проектами (Issue tracking system);
- CRM – Система управления взаимодействием с клиентами;
- система управления и контроля над разработкой SW решений (SW development tracking and supervising tool, bugtracking);
- система управления и контроля над проектами и т.д.

СОО обеспечивает в режиме реального времени отслеживание закрепленных за ответственными лицами задач, что делает ее незаменимой для руководителей при организации эффективного труда сотрудников, а также при организации взаимодействия сотрудников между собой.

СОО позволяет работать со справочниками данных. Справочники программы, используются для исключения заполнения полей вручную. Из справочника могут быть взяты следующие данные:

- статус заявки;
- автор записи;
- тип записи;
- название подсистемы;
- приоритет;
- дата обнаружения;
- тип обращения;
- срок исполнения заявки.

Использование справочников позволяет избежать ошибок при наборе реквизитов заявки (задачи), так как в справочниках хранится достоверная информация по заполнению полей в заявке.

Задача регистрируется в рамках работы структурных подразделений разработчика ПО и линии технической поддержки пользователей ПО. Это может быть задача от руководства, связанная с доработкой по требованиям заказчика; ошибка, выявленная при тестировании ПО; проблема, возникшая у пользователя или сообщение об ошибке от системы, обнаруженное пользователем в процессе эксплуатации ПО. Так же задача может возникнуть в рамках работы группы разработки ПО или анализа ПО при его согласовании с заказчиком продукта.

Далее, созданная задача поступает на рассмотрение руководителей отделов. Исходя из контекста, они принимают решение об отказе или назначении в работу поступившей задачи.

Принятая в работу задача поступает в группу тестирования для подтверждения факта обнаруженной ошибки или непосредственно в группу разработки, если такое подтверждение не требуется. Если ошибка не будет подтверждена в результате тестирования, она отклоняется. Подтвержденная тестировщиками ошибка поступает для решения непосредственным разработчикам программного продукта.

После доработки или разработки вновь задача возвращается на тестирование, и если будет подтверждено исправление обнаруженной ошибки или будет получена корректная работа созданного компонента, задача закрывается.

## **Общий алгоритм обновление ПО**

Обновления изделия, если при поставке обновлений не обговорено иное, осуществляются согласно условиям договора о поставке изделия и технического сопровождения в следующих вариантах:

- Производится полная деинсталляция текущей версии ПО с последующей установкой и настройкой обновленного ПО в полном соответствии с программной документацией;
- Производится ручное обновление ПО (администратором системы) с использованием

сервисов цифровой дистрибуции.

## Порядок установки критических обновлений

При возникновении необходимости установки критических обновлений производитель незамедлительно осуществляет доведение до потребителей информации о необходимости обновления и предоставляет возможность его получения по цифровому каналу.

Потребитель при получении указанной информации осуществляет получение обновления и незамедлительно применяет его, о чем делает соответствующую отметку в эксплуатационной документации с указанием типа, даты и времени обновления, а также с указанием фамилии лица, применившего его.

## Характеристики репозитория изделия

Репозиторий изделия служит хранилищем дистрибутивных пакетов изделия, обеспечивает сервис онлайн-дистрибуции изделия для потребителей. Физически репозиторий представляет собой набор сетевых ресурсов в сети передачи данных с идентифицированными для пользователей сетевыми именами. Идентификация доступного репозитория в сети передачи данных организуется на уровне сетевых протоколов.

Функционирование сервисов репозитория изделия в сети передачи данных осуществляется по следующим сетевым протоколам и портам:

- протокол HTTPS, порт 8443. Данный порт зарезервирован для обеспечения аутентификации и построения защищенного TLS-соединения на основании сертификата открытого ключа пользователя с использованием инфраструктуры PKI и на данном порту обеспечивается доступ потребителей к ресурсам репозитория и получение обновлений;

Функции репозитория изделия заключаются в обеспечении следующих сервисов:

- сервис онлайн-дистрибуции изделия;
- сервис обновления изделия;
- сервис верификации изделия.

Сервис онлайн-дистрибуции обеспечивает полную идентичность бинарных образов изделия, представленных на физических цифровых носителях информации (флеш-накопители, CD/DVD диски и т.д.), и бинарных образов изделия, размещаемых в ресурсах репозитория производителя и загружаемых потребителями по сети передачи данных.

Сервис обновления изделия круглосуточно обеспечивает обновление изделия для конечных потребителей изделия.

Сервисы онлайн-дистрибуции и обновления изделия обеспечивают для конечных потребителей однозначную идентификацию изделия по версиям и релизам.

Верификация изделия обеспечивается для конечных потребителей, для установления подлинности изделия, его бинарных файлов и цифровых установочных образов дистрибутивов изделия. Для установления подлинности изделия проводится автоматизированный подсчет контрольных сумм бинарных файлов и установочных образов изделия.

## Способы распространения изделия

Дистрибутив(ы) ПО актуальной версии (стадия «Релиз») размещаются разработчиком на сайте, становятся доступными для загрузки после авторизации пользователя в его личном кабинете.

Так же по запросу пользователя ему могут быть высланы ссылки на скачку дистрибутивов предыдущих версии ПО различных стадий.

Поставка неисключительных прав (лицензий) на использование того или иного функционала ПО осуществляется по договору (акту, соглашению) о передаче неисключительных прав между потребителем и производителем. Данные о лицензиях, в том числе технологические, необходимые для их применения, могут быть запрошены через задачу в личном кабинете или электронным письмом на адрес [info@nord-m.com](mailto:info@nord-m.com) и передаются потребителю в электронном виде.

---

## Описание штатной структуры проекта

Руководство проектом осуществляет технический директор ООО «Норд-М».

Штатная структура проекта включает:

- Отдел разработки программных средств;
  - Отдел разработки аппаратных средств;
  - Отдел тестирования;
  - Отдел технической поддержки;
- 
-