

????? ?????????????? Printum: ??? ??????? ???????

?????? ?????????????? ??????? ?????????? ??

Инженеру необходимо научиться читать архитектурную схему так, чтобы видеть потоки данных: как задание, атрибуты или событие проходят через компоненты системы.

Понимание потока данных, позволяет:

- диагностировать проблему в проекте любой сложности, а не только в тех конфигурациях, которые инженер уже видел;
- понимать, зачем нужен каждый компонент и что случится, если он откажет;
- объяснять заказчику и коллегам, почему архитектура устроена именно так, а не иначе.

Схема Printum описывает не одну конфигурацию, а семейство из четырёх архитектурных блоков, которые комбинируются между собой в зависимости от масштаба и требований инфраструктуры заказчика. Разбор каждого блока — отдельная статья цикла:

1. Обзор архитектуры и легенда схемы (эта статья).
2. Система мониторинга.
3. ПринтМенеджер в базовой конфигурации: Сингл.
4. Отказоустойчивость и горизонтальное масштабирование.
5. Филиальная сеть.

?? Printum

Перед дальнейшим чтением откройте архитектурную схему Printum. В этой статье мы будем ссылаться на её элементы и условные обозначения.

→ [Скачать архитектурную схему Printum](#)

Если абстрагироваться от внутреннего устройства, Printum — комплексное решение из двух веб-приложений и набора вспомогательных программ и сервисов:

- **Система мониторинга принтеров и МФУ** — ядро системы. Собирает и агрегирует данные из LDAP-каталогов (доменов), от сетевых принтеров и МФУ, с локальных устройств (подключенных по USB) и от ПринтМенеджеров.
- **Система управления печатью (ПринтМенеджер)** — обрабатывает задания печати, копирования и сканирования, управляет правилами и авторизацией пользователей на МФУ.

Функциональные и нефункциональные требования

Прежде чем разбирать компоненты, полезно разделить требования к системе на два типа — это объясняет, зачем часть архитектуры существует.

Функциональные требования — что система должна делать: печать (прямая и отложенная, с клиентом ПринтМенеджера и без), копирование, сканирование (в почту, в сетевую папку), авторизация пользователя на устройстве, сбор данных о состоянии парка устройств, управление пользователями и правилами.

Нефункциональные требования — какими свойствами система должна обладать при выполнении этих функций: отказоустойчивость (продолжать работать при потере части инфраструктуры), масштабируемость (выдерживать рост числа устройств, пользователей, заданий), производительность, безопасность (не хранить данные в открытом виде, разграничивать доступ и т.п.).

Часть компонентов на схеме существует не ради функций, а ради нефункциональных требований — например, разделение на PostgreSQL и ClickHouse существует ради производительности, а балансировщик HAProxy и Redis Sentinel — ради отказоустойчивости и масштабируемости. Если непонятно, зачем нужен компонент, полезно задать вопрос: он выполняет функцию или обеспечивает нефункциональное требование?

??????? ?????

Архитектурная схема Printum использует ограниченный набор условных обозначений. Разобравшись с ними один раз, любой блок схемы читается одинаково.

Обозначение	Значение
Зелёный цилиндр	База данных. Под каждым цилиндром подписан конкретный движок: PostgreSQL, ClickHouse, Redis.
Синий прямоугольник	Python-приложение — бэкэнд одной из систем (Django, Celery, планировщик).
Синий прямоугольник со скруглёнными краями	Обратный прокси / веб-сервер (nginx) или балансировщик (HAProxy). Устанавливает соединения между компонентами, браузером и личным кабинетом.

Обозначение	Значение
Пунктирный прямоугольник — архитектурный блок	Какие компоненты необходимы для выполнения определённой функции. Отвечает на вопрос «из чего это состоит».
Пунктирный прямоугольник — логический блок	Где компоненты физически размещаются в инфраструктуре (на каком сервере, в каком сегменте сети). Отвечает на вопрос «где это стоит».
Прочие значки	МФУ, АРМ пользователя, внешние системы (домен, SMTP/IMAP, SIEM) — обозначаются отдельными иконками и подписываются явно.

Различие между архитектурным и логическим блоком важно: один и тот же архитектурный блок (например, «ПринтМенеджер») может быть развёрнут в нескольких логических местах — в филиале А, в филиале Б, — и в каждом из них будет одинаковый набор внутренних компонентов.

?????? ?????????????????? ??????
????????? ????????

Полная схема Printum состоит из четырёх архитектурных блоков. На практике в конкретном проекте одновременно присутствует блок мониторинга (всегда один) и один или несколько блоков управления печатью — в зависимости от масштаба и требований к отказоустойчивости.

- 1. **Система мониторинга** — единственный экземпляр в инфраструктуре независимо от количества филиалов или ПринтМенеджеров. Подробности — в статье 2.
- 2. **ПринтМенеджер в базовой конфигурация (Сингл)** — типовая установка: один сервер обрабатывает печать/сканирование/копирование для одного офиса или сегмента сети. Подробности — в статье 3.
- 3. **ПринтМенеджер с горизонтальным масштабированием (Кластер)** — несколько ПринтМенеджеров работают как единый кластер через балансировщик HAProxy. Применяется, когда одного сервера не хватает по нагрузке или когда заказчику нужна отказоустойчивость обработки заданий. Подробности — в статье 4.
- 4. **Филиальная сеть** — несколько независимых ПринтМенеджеров в разных географических точках, каждый обслуживает свой филиал и синхронизируется с общим сервером мониторинга. Применяется при значительных сетевых задержках между офисами или в тех случаях, когда простой сервиса печати при обрыве связи с центральным сервером критичен. Подробности — в статье 5.

Блоки 3 и 4 не взаимоисключающие: в одном из филиалов филиальной сети вполне может быть развёрнута схема с балансировкой, если нагрузка именно этого филиала того требует.

????? ??????? ?? ????????? ????????:
???????????? ????????

Прежде чем переходить к деталям каждого компонента, полезно держать в голове базовый маршрут задания печати — вне зависимости от конкретной конфигурации он всегда проходит одну и ту же последовательность звеньев:

АРМ пользователя → драйвер виртуального принтера → ПринтМенеджер (обработка атрибутов и образа) → МФУ (авторизация и печать)

Статистика по обработанным заданиям после этого возвращается из ПринтМенеджера в систему мониторинга (по расписанию или мгновенно). Данные же о состоянии самих устройств идут отдельным маршрутом — от МФУ через сетевой агент напрямую в мониторинг, минуя ПринтМенеджер.

Это два разных потока: поток заданий (через ПринтМенеджер) и поток данных о состоянии устройств (через агентов мониторинга). Их не стоит путать при диагностике — если не печатает конкретное задание, проблема всегда в первом потоке; если в личном кабинете не видно актуального состояния принтера (уровень тонера, статус ошибки), проблема — во втором.

??? ??? ????????? ? ?????????????????

Способность назвать компонент по картинке — не цель. Цель — при жалобе «не работает печать» уметь мысленно провести задание по цепочке АРМ → драйвер → ПринтМенеджер → МФУ и определить, на каком звене данные потерялись. Подробно эта методика разобрана в отдельной статье [Модель диагностики](#) — она построена именно на знании этой цепочки, которое даёт архитектура.

Revision #1

Created 2026-07-12 14:49:36 UTC by DD

Updated 2026-07-12 14:56:17 UTC by DD