Впервые освещена в одноимённой книге ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *Lean Software Development*) Мэри Поппендик и Toма Поппендика. В книге представлены традиционные принципы бережливого производства применительно к разработке программного обеспечения, также набор из 22 инструментов (практик) и их сравнение с гибкой методологией разработки. Мэри и Том участвовали в ряде различных конференций, посвящённых методикам [Agile](https://ru.wikipedia.org/wiki/Agile" \o "Agile), что объясняет известность концепции бережливого производства среди сообщества гибкой методологии разработки

* **Исключение потерь**. Потерями считается всё, что не добавляет ценности для потребителя. В частности: излишняя функциональность; ожидание (паузы) в процессе разработки; нечёткие требования; бюрократизация; медленное внутреннее сообщение.
* **Акцент на обучении.** Короткие циклы разработки, раннее тестирование, частая [обратная связь](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%80%D0%B0%D1%82%D0%BD%D0%B0%D1%8F_%D1%81%D0%B2%D1%8F%D0%B7%D1%8C) с заказчиком.
* **Предельно отсроченное принятие решений.** Решение следует принимать не на основе предположений и прогнозов, а после открытия существенных фактов.
* **Предельно быстрая доставка заказчику.** Короткие итерации.
* **Мотивация команды.** Нельзя рассматривать людей исключительно как ресурс. Людям нужно нечто большее, чем просто список заданий.
* **Интегрирование.** Передать целостную информацию заказчику. Стремиться к целостной архитектуре. [Рефакторинг](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D1%84%D0%B0%D0%BA%D1%82%D0%BE%D1%80%D0%B8%D0%BD%D0%B3" \o "Рефакторинг).
* **Целостное видение.** Стандартизация, установление отношений между разработчиками. Разделение разработчиками принципов бережливости. «Мыслить широко, делать мало, ошибаться быстро; учиться стремительно».

Некоторые практики бережливой разработки аналогичны практикам быстрой разработки, а некоторые несколько различаются. Примеры практик:

* Обнаружение потерь («[en:Muda (Japanese term)](https://en.wikipedia.org/wiki/Muda_(Japanese_term)" \o "en:Muda (Japanese term))»)
* [Систематизирование потока ценности](https://ru.wikipedia.org/w/index.php?title=%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0%D1%82%D0%B8%D0%B7%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5_%D0%BF%D0%BE%D1%82%D0%BE%D0%BA%D0%B0_%D1%86%D0%B5%D0%BD%D0%BD%D0%BE%D1%81%D1%82%D0%B8&action=edit&redlink=1) (Value stream mapping)
* [Теория ограничений](https://ru.wikipedia.org/wiki/%D0%A2%D0%B5%D0%BE%D1%80%D0%B8%D1%8F_%D0%BE%D0%B3%D1%80%D0%B0%D0%BD%D0%B8%D1%87%D0%B5%D0%BD%D0%B8%D0%B9)
* [«Вытягивающая»](https://ru.wikipedia.org/wiki/%D0%92%D1%8B%D1%82%D1%8F%D0%B3%D0%B8%D0%B2%D0%B0%D1%8E%D1%89%D0%B5%D0%B5_%D0%BF%D1%80%D0%BE%D0%B8%D0%B7%D0%B2%D0%BE%D0%B4%D1%81%D1%82%D0%B2%D0%BE) система ([Канбан](https://ru.wikipedia.org/wiki/%D0%9A%D0%B0%D0%BD%D0%B1%D0%B0%D0%BD_(%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B0)" \o "Канбан (разработка)))
* [Теория массового обслуживания](https://ru.wikipedia.org/wiki/%D0%A2%D0%B5%D0%BE%D1%80%D0%B8%D1%8F_%D0%BC%D0%B0%D1%81%D1%81%D0%BE%D0%B2%D0%BE%D0%B3%D0%BE_%D0%BE%D0%B1%D1%81%D0%BB%D1%83%D0%B6%D0%B8%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F)
* [Мотивация](https://ru.wikipedia.org/wiki/%D0%9C%D0%BE%D1%82%D0%B8%D0%B2%D0%B0%D1%86%D0%B8%D1%8F)
* Измерения