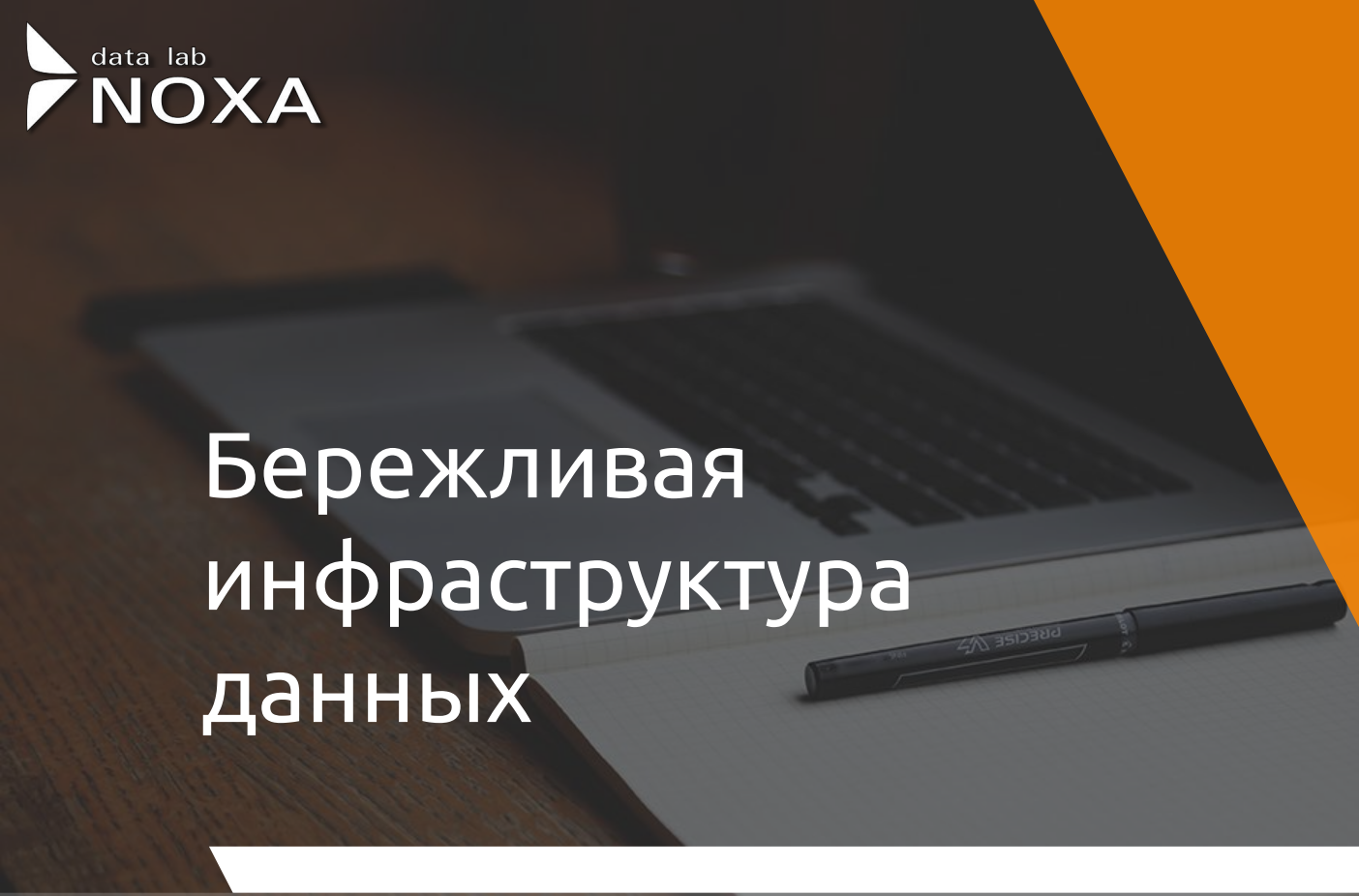
data lab
NOXA

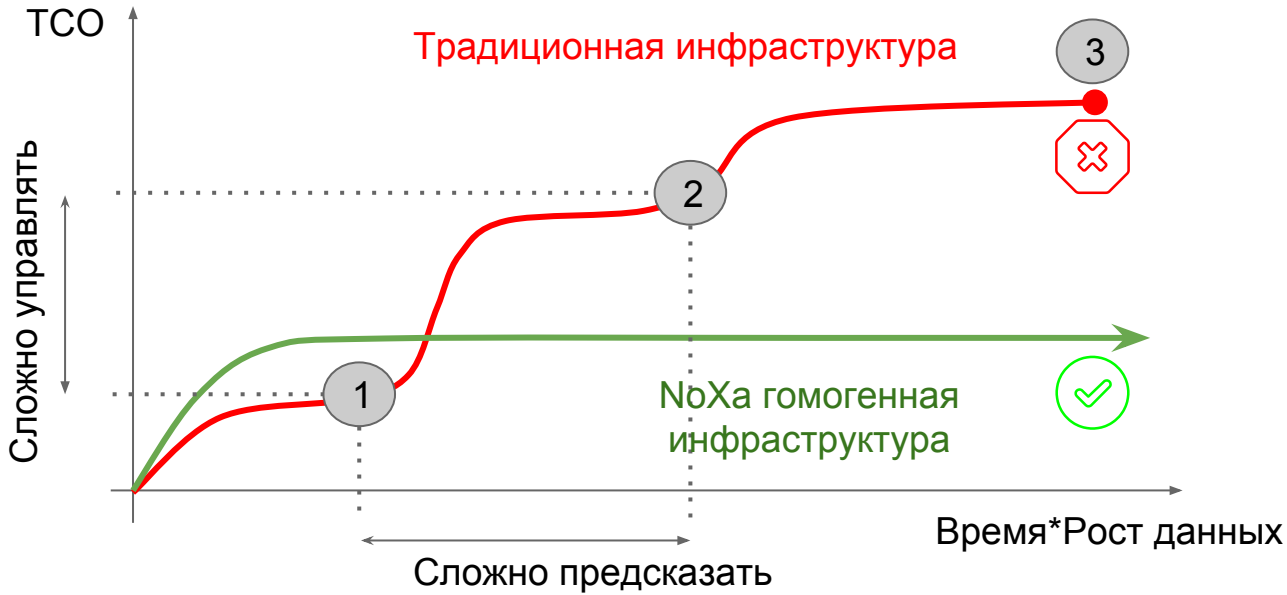
Бережливая инфраструктура данных



Вызовы с которыми сталкивается бизнес

- TCO растёт непредсказуемо
- Управление рисками осложнено
- Данные убивают бизнес

- NoXa просто работает



Структура рисков

1. OLAP\OLTP противоречие - схемы данных должны быть разделены. “Чистота” транзакций теряется. Потеря доступности данных.

2. Обеспечение единой точки правды требует дорогостоящих инструментов MDM, а также интеграционных инструментов Data Quality и ETL. Сложный многослойный дизайн.

3. Данные порождают данные. Управление единой точкой правды - невозможно, ACID не поддерживается. Эффективное моделирование невозможно. ETL \ Replication порождает избыточные сущности. Бизнес тратит всё больше и больше денег на поддержание инфраструктуры. Требуются технологии BIG DATA.

РСУБД пока справляются

Требуются Data Quality, MDM, ETL\Replication и Database Appliance.

Инфраструктура становится “Озером Данных” с грязной водой

Концепция Smart Data

- *Данные должны быть реляционными*
- *Данные должны быть виртуально централизованы*
- *Реляционная модель должна быть нормализована*
- *Данные должны быть “надежны”*
- *Данные должны быть транзакционны*
- *Производительность и доступность должны быть под контролем, стоимость обеспечения должна быть предсказуема*
- *Объемы не должны иметь значения, данные должны быть дружественны к интеграции*

7 нормальная форма – основа качества данных

- **Высшая, теоретически возможная степень нормализации**

Позволяет проектировать структурированные модели без аномалий.

- **“Insert only” физическое хранилище на основе SQL СУБД**

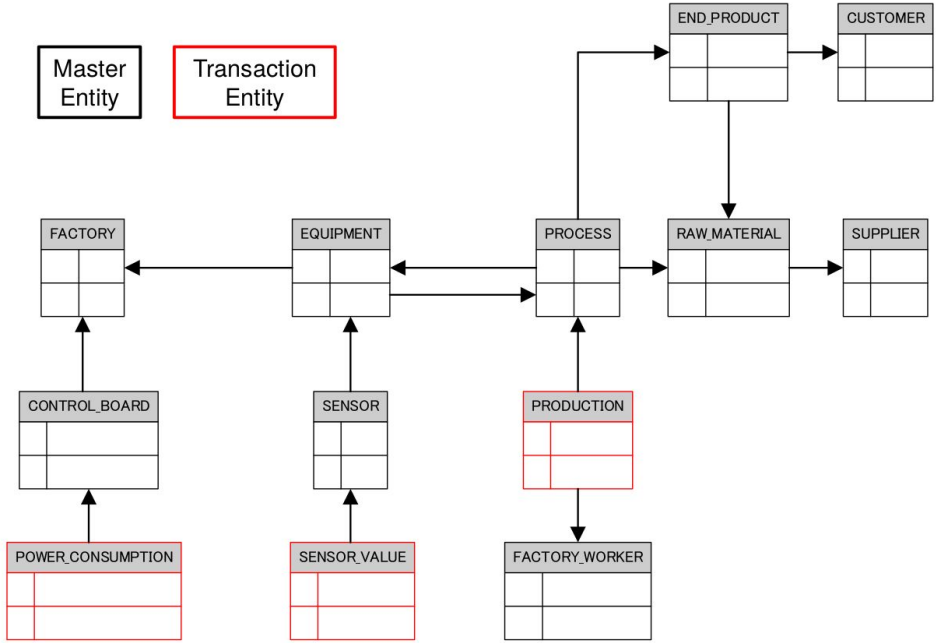
Позволяет использовать подходы HPC и MPP за счет минимизации блокировок, в том числе и на аппаратном уровне.

- **Все сущности реплицированы и организованы как SHARED NOTHING кластер**

Высокопроизводительные партицированные выборки “SELECT” из индексированного хранилища с “JOIN” и “FORK JOIN” агрегации в памяти.

- **Единая точка правды - гарантирована**

Физическое разделение МАСТЕР и ТРАНЗАКЦИОННЫХ данных



7 нормальная форма – залог продуктивности

- **Методология моделирования**

Определяет правила и лучшие практики декомпозиции бизнес требований. Определяет практику и процессы разработки.

- **Отсутствие WAL\REDO журналирования**

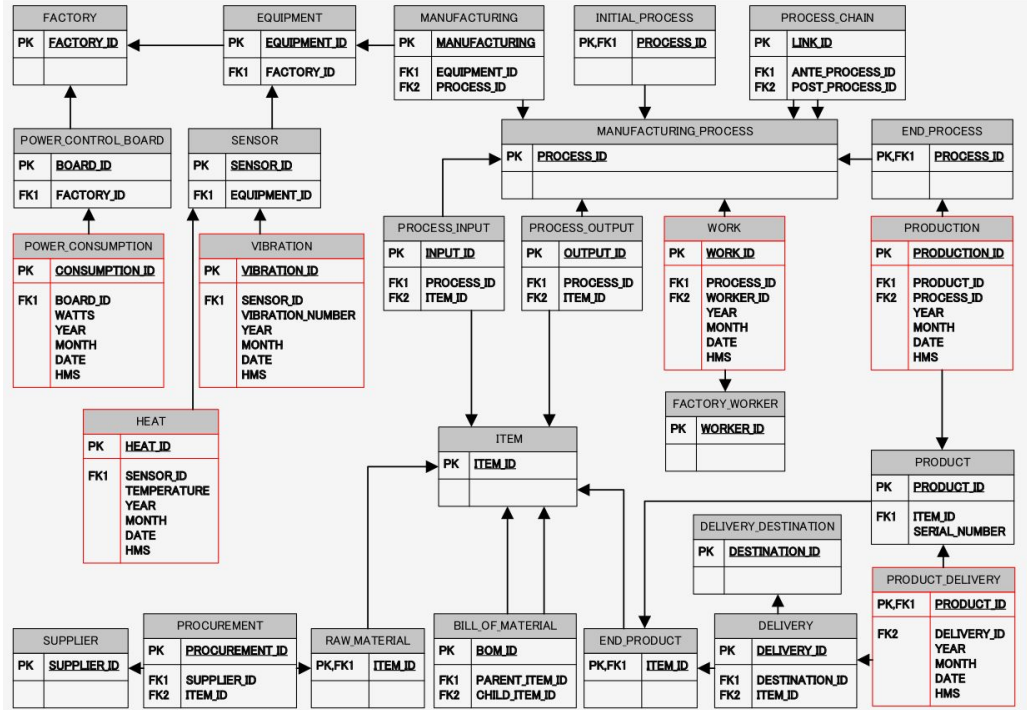
HPC&MPP подходы без блокировок даже на низком IO уровне.

- **Простота и элегантность**

Ключ/Значение хранение.

- **Логические Updates и Delete**

Фактически реализованы как вставка.



Избыточность - основа надежности и производительности.

- **Логические изменения распределяются как autocommit микро транзакции**

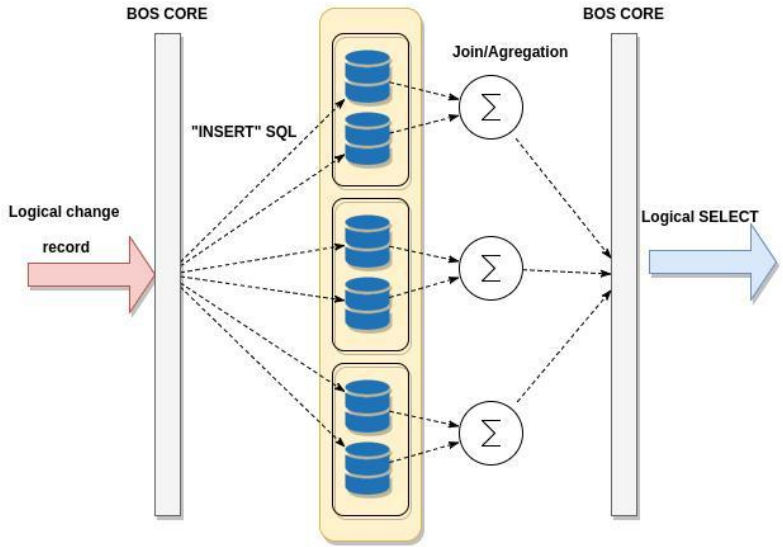
Не требуется синхронизированная репликация. Нет FSYNC, нет блокировок, обработка в реальном времени.

- **Избыточность 6x в базовой инсталляции**

Позволяет достичь 6 кратного увеличения производительности выборок по сравнению с обычными СУБД.

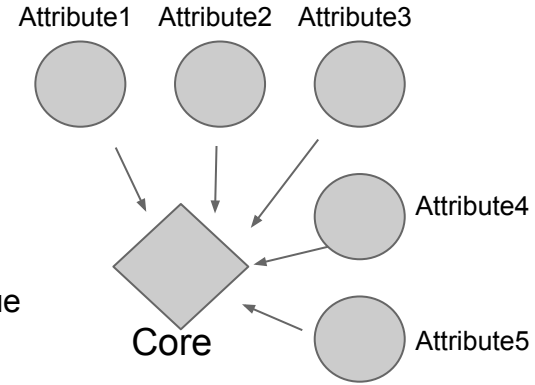
- **Отсутствие единой точки отказа**

6 кратная избыточность обеспечивает надёжность хранения.



ACID и MVCC определяется приложением по требованию во время выборки

- **Бизнес приложение оперирует представлениями (views).**
- **ACID реализовано связыванием CORE ID, играющим роль XID в RDMS**



Insert algorithm:

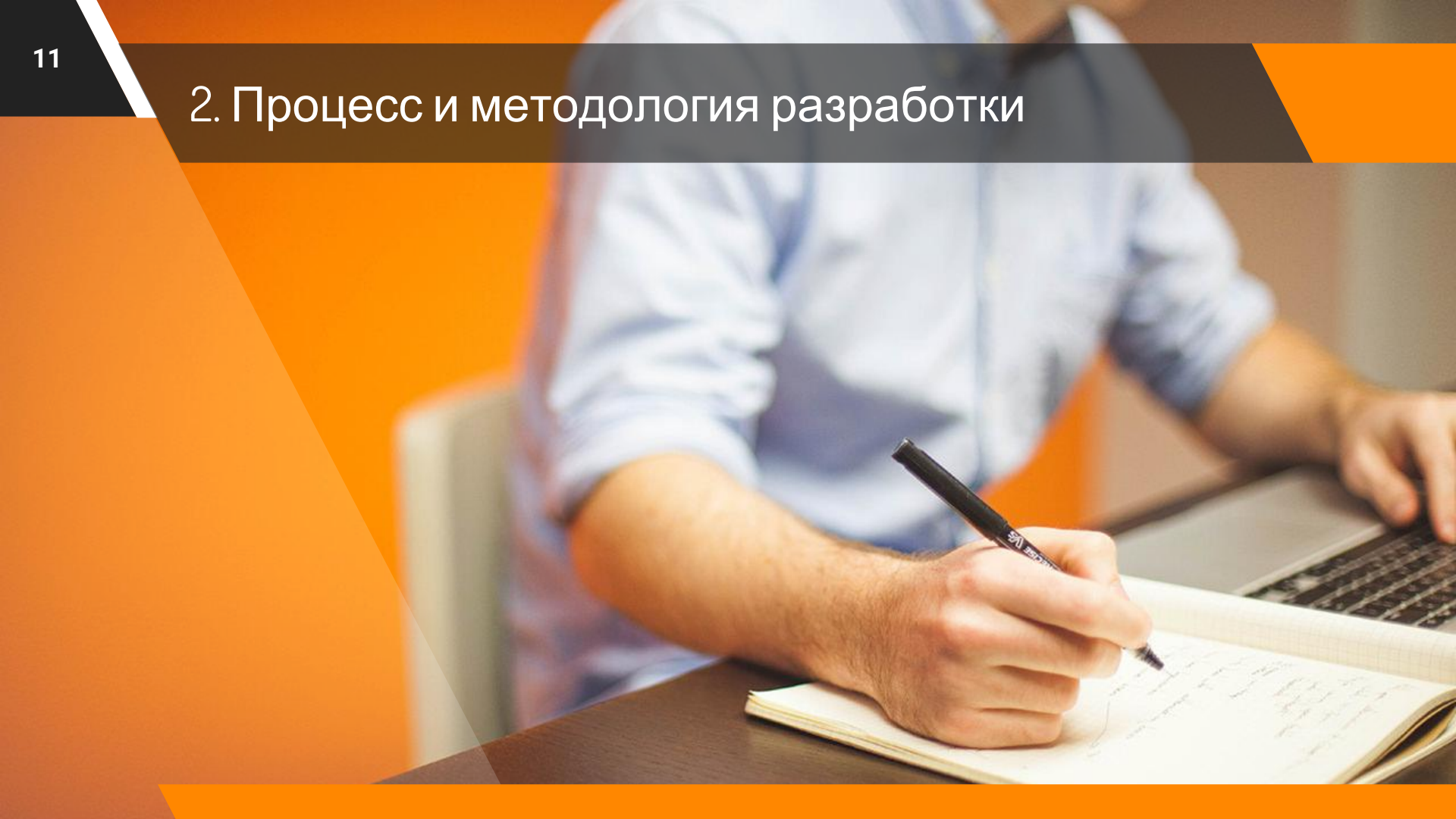
1. *Prepare CORE ID with definite timestamp.*
2. *Insert In Attribute 1 tables.*
3. *Insert In Attribute 2 tables.*
4. *Insert In Attribute 3 tables.*
5. *Insert In Attribute 4 tables.*
6. *Insert In Attribute 5 tables.*
7. *Insert in CORE ID tables.*

Если что то пошло не так, в таблицу CORE ID никогда не произойдет вставка, как следствие никогда не произойдёт выборка результирующего кортежа по ключу CORE ID. Этот эффект является логическим Roll back. Если вставка произошла нормально, значение доступно для выборки по CORE ID.Это логический Commit.

NoXA

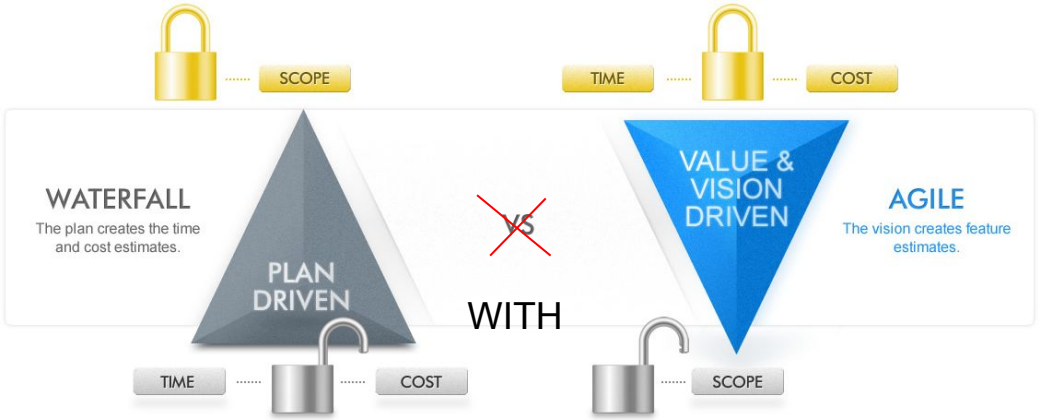
First in the word, brings the state of the art data infrastructure where ACID is guaranteed among 10k> distributed tables during SELECT rather than TX Logging, Distributed XA TX, and multiphase commit.

2. Процесс и методология разработки



Лучшее из 2х миров

- Короткие циклы итераций.
- Точное планирование выпусков.
- Тесная интеграция с бизнесом.
- Sustainable agility архитектуры, дизайна и модели данных.
- Лучшие практики SE.



Java is the CORE, but BOS is more than Java

- **JPA alike Java 1.8 SDK**

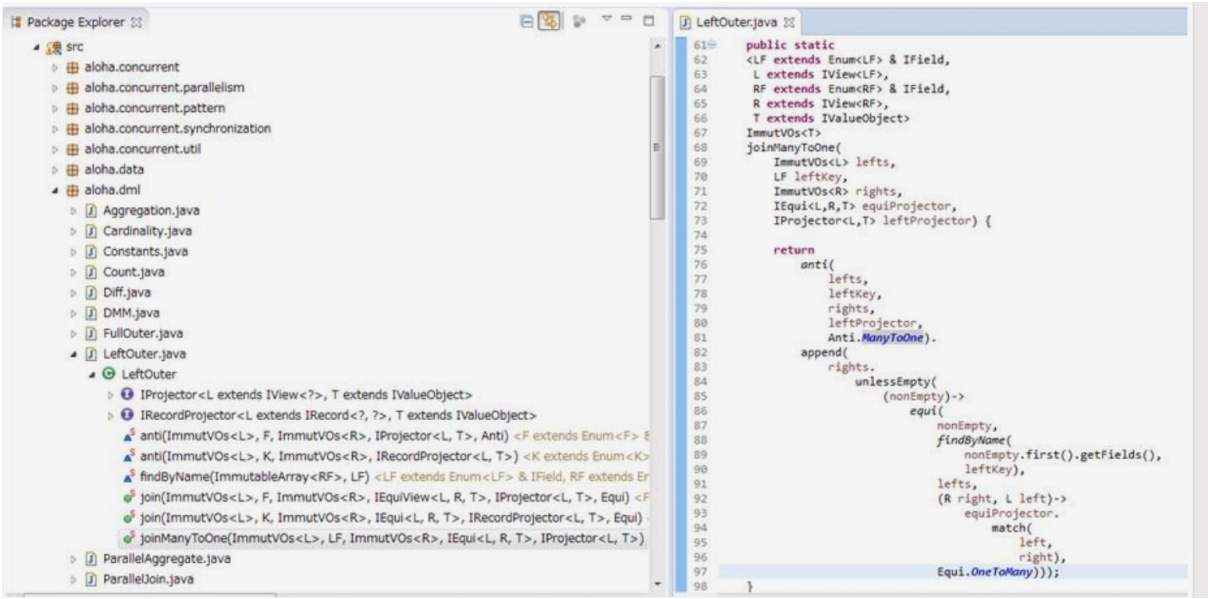
Все сущности реализованы как отдельные типы (generics).

Реляционная модель верифицируется во время компиляции.

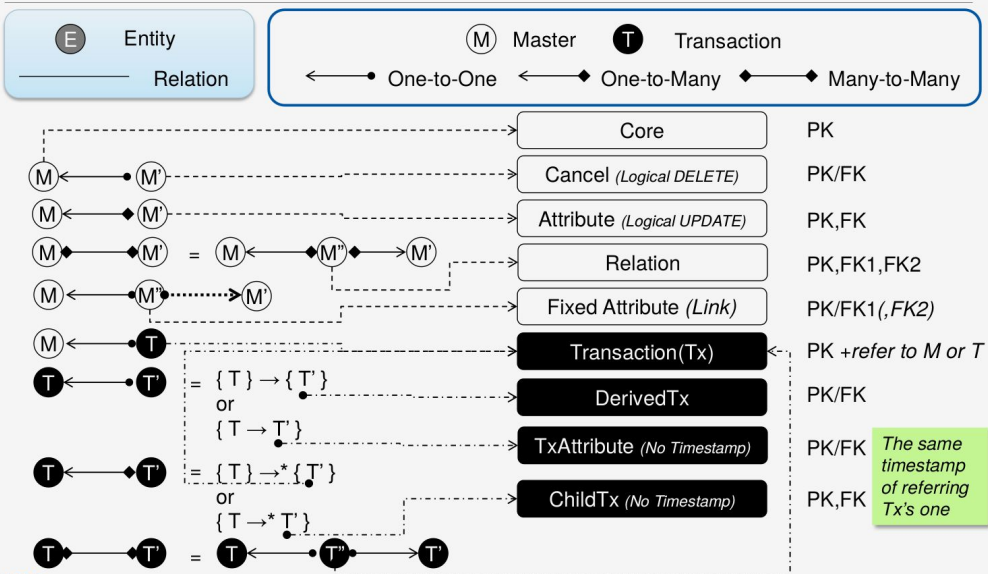
- **Unit test free**

BOS SDK ориентирован на производительность разработчиков.

- **Нет локальных переменных, всё IMMUTABLE**



Методология Data Oriented Approach



ICoreEntity.java

```
1  package aloha.module.object.record
2
3  import aloha.module.object.record.
5
6  public interface ICoreEntity
7      <C extends Enum<C> & IColumn,
8       K extends Enum<K> & IKey>
9      extends IPivot<C,K> {
```

IRelationEntity.java

```
1  package aloha.module.object.record.master;
2
3  import aloha.module.object.record.IColumn;
5
6  public interface IRelationEntity
7      <C extends Enum<C> & IColumn,
8       K extends Enum<K> & IKey,
9       P extends ICoreEntity<?,?>,
10      S extends ICoreEntity<?,?>>
11     extends
12         IRelation<C,K>,
13         aloha.module.object.record.IRelationEntity<P,S> {
```

IAttributeEntity.java

```
1  package aloha.module.object.recor
2
3  import aloha.module.object.record
6
7  public interface IAttributeEntity
8      <C extends Enum<C> & IColumn,
9       K extends Enum<K> & IKey,
10      E extends ICoreEntity<?,?>>
11     extends
12         ICoreAttribute<C,K>,
13         IReferenceEntity<E> {
```

```
public interface IHierarchyEntity
    <C extends Enum<C> & IColumn,
     K extends Enum<K> & IKey,
     E extends ICoreEntity<?,?>>
    extends
        IRelationEntity<C,K,E,E>,
        IReferenceEntity<E> {
```

BOS&DOA

- **Бизнес логика реализована в Java**

IConceptual

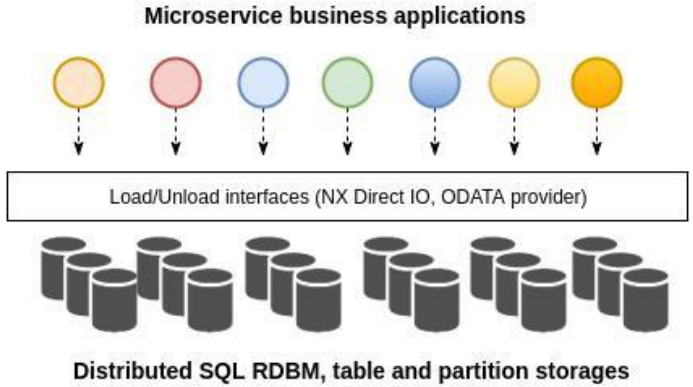
IGateWay

IActor

IMDMlookUp

- **Микросервисная архитектура**

Гибридный OLAP\OLTP, витрины данных, real time lookup, и многие другие сервисы, гарантированно разделяют перзистивный слой



3. Что такое NoXA?

- Noxa - technology enabler на восточноевропейском рынке.
- Noxa разработчик проприетарного ПО в BOS экосистеме.

Products and services

- **Noxa Data Management Console**

Graphical User Interface that gives the control of all aspects of infrastructure.

- **Noxa Direct IO**

The parallel load, unload utility and drivers.

- **Noxa ODATA integration**

- **Project supervision**

Consulting and PM services

- **R&D services**

We build and start brand new systems

- **Training**

Спасибо!

amergasov@noxa-datalab.com

www.noxa-datalab.com