

Work with InterSystems.

Not separate systems.



InterSystems Caché. Промышленная СУБД для Web 2.0

Database as a Service

oleg.olenin@intersystems.com

INTERSYSTEMS

В следующие 45 минут



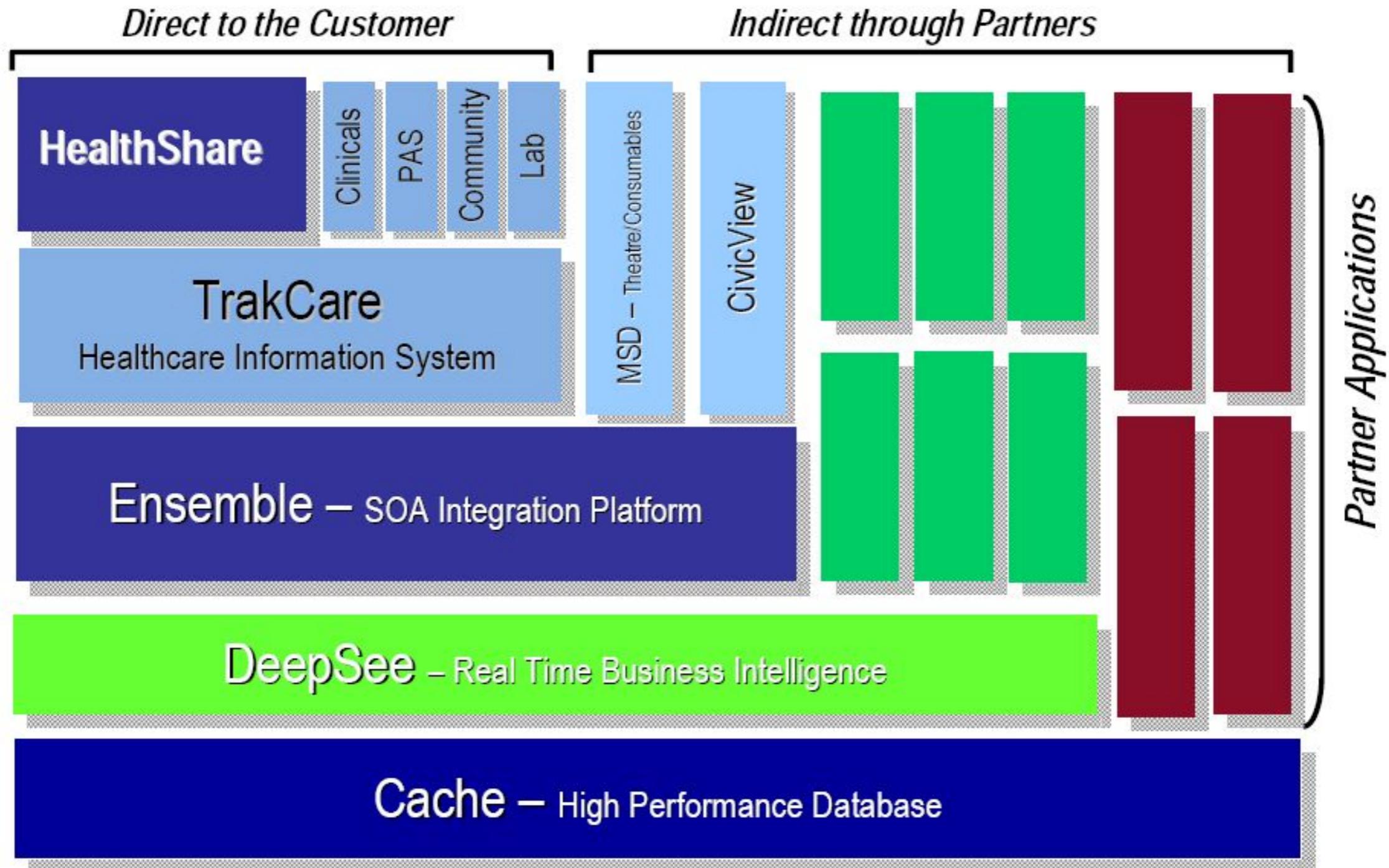
- Что нового в InterSystems Cache?
- Приложения эпохи Web 2.0 и кризис традиционного подхода к работе с СУБД
- InterSystems Caché как Application Database
- InterSystems Caché в инфраструктуре Web 2.0
- Caché и Enterprise 2.0
- Что дальше? Требования к СУБД в эпоху Web 3.0

Успешная компания с 1978 года



- Штаб-квартира – Кембридж, Массачусетс, США
- Постреляционная СУБД Caché + платформа для интеграции Ensemble + платформа для обмена медицинской информацией HealthShare
- Оборот компании стабильно растет
- Офисы в 22 странах мира

Продукты InterSystems



Продукты InterSystems



Высокопроизводительная СУБД

Быстрая разработка
приложений



Быстрая интеграция приложений

Быстрая разработка
композиционных приложений

InterSystems. Что нового?



- В прошлом году исполнилось 30 лет компании
- Исполнилось 10 лет и филиалу в России
- Gartner опять назвал Ensemble лучшим продуктом среди интеграционных платформ для композитных приложений
- Вышел InterSystems DeepSee - embeded, realtime, actionable BI
- Выходит Cache 2009.1
 - Еще быстрее
 - Еще более открытая
 - .Net и Java Gateways

Приложения Web 2.0



- Интернет в режиме Read Write
 - приложения в своей работе используют информацию, предоставляемую и создаваемую пользователями
 - уникальный пользовательский контент и сообщество становятся конкурентным преимуществом - воспроизвести функциональность приложения можно, данные и пользователей - нет
- Доставка и развертывание приложений
 - сеть как платформа
 - повторное использование специализированных интернет сервисов
 - software as a service
 - различные устройства доступа

Требования к СУБД



- Природа данных претерпела изменения
- Взаимодействие с приложением - смена приоритетов
- СУБД не должна снижать скорость изменений, вносимых в приложения или делать ее непредсказуемой
- Готовность к большому взрыву в случае лавинообразного роста пользователей
- Постоянная стоимость за ресурс. Easy come - easy go.



- Природа данных эпохи Web 2.0 претерпела изменения, которые делают РСУБД менее эффективными
 - отказ от контроля модели данных средствами СУБД - либо данные достаточно просты, пример с MySQL
 - модель не укладывается в табличное представление и требуется логика приложения для ее дополнительной интерпретации
- Модель предметной области
 - работа с полуструктурированными данными
 - широкое использование информационных объектов, имеющих иерархическую структуру (документы, сообщения, DOM)
 - выбор графового представления как естественного

Application Database vs. Integration Database



- Мартин Фаулер о корпоративных приложениях и СУБД
 - Application Database - одно приложение, одна БД
 - Логика и метаданные целиком в приложении. СУБД не разделяемый ресурс
 - Контроль данных в приложении, СУБД концентрируется на инфраструктурных задачах
 - Доступ всегда через сервисы, API
 - Integration Database - много приложений, одна БД
 - Логика и метаданные в БД
 - Контроль данных в СУБД, двойная работа

Caché как Application Database для Web 2.0

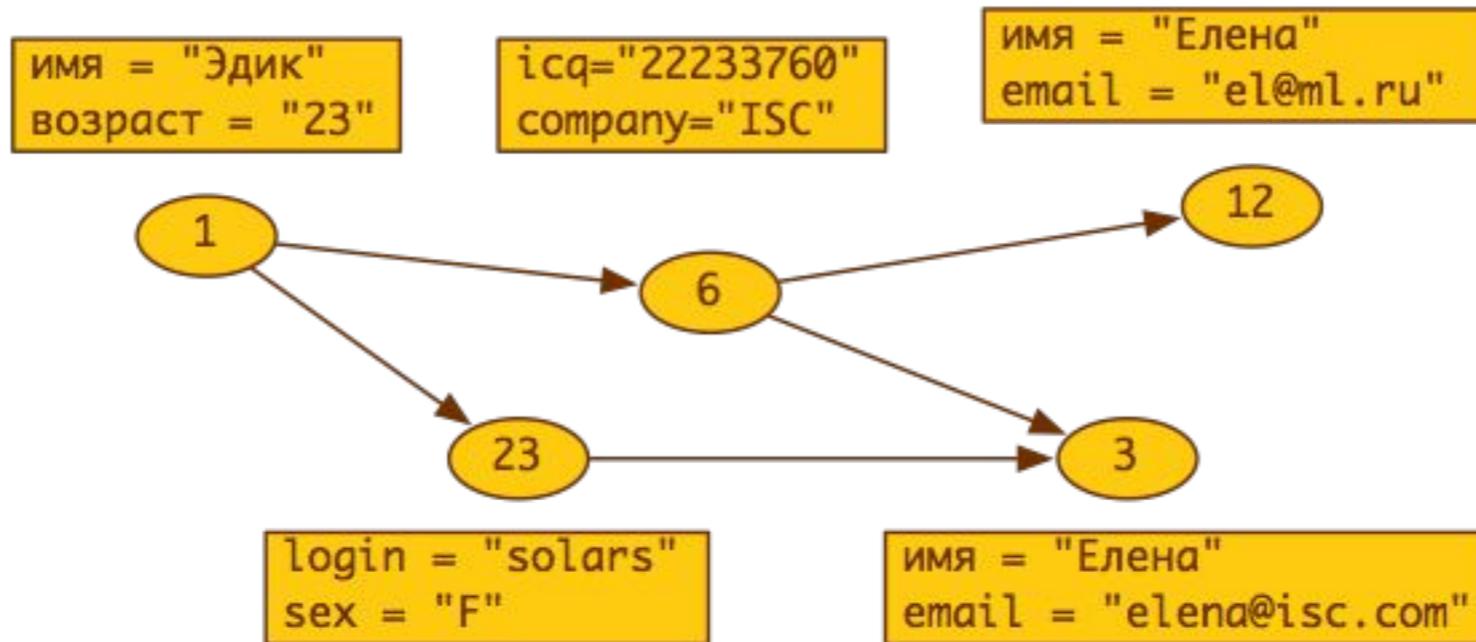


Caché как Application Database для Web 2.0



- Модели данных
 - Caché - готовая платформа для разработки специализированных хранилищ, устойчивых к нагрузке и масштабируемых
 - позволяет сосредоточиться разработчикам на создании СУБД со своей, характерной, моделью данных
 - при этом обеспечивает полностью все необходимые инфраструктурные элементы для такой заказной СУБД
 - Caché - три вида моделей представления
 - Прямой - ассоциативные массивы (глобалы) и узлы
 - Объектный - классы и объекты
 - Реляционный - таблицы и записи

Квазиструктурированные и графовые объекты в Cache



Представление и сохранение графа и объектов

Cache

```
s ^node(1,"имя") = "Эдик"
s ^node(1,"возраст") = 23

s ^node(23,"login")="solars"
s ^node(23,"sex")="F"
Варианты описания связей:
s ^node(1,"link",23) = "земляк"
или
s ^node(1,"link",23,"login")="solars"
s ^node(1,"link",23,"sex")="F"
или
s ^node(1,"links") = "23,6"
```

JSON

```
{ "node":
  { "id": 1,
    "имя": "Эдик",
    "возраст": 23,
    "node":
      { "id": "23",
        "login": "solars",
        "sex": "F"
      }
    ....
  }
}
```

Скорость изменений: чем проще, тем лучше



- Работа с глобалами
 - глобалы Caché
 - **`set ^account("olenin","pwd") = "*****"`**
 - так могло быть в PHP или Javascript
 - в Javascript скоро так и будет
- Глобалы позволяют хранить и обрабатывать полуструктурированные данные, в том числе объекты с переменным количеством атрибутов
- Могут быть использованы для представления данных, имеющих иерархическую природу, например, навигации, структуры форумов, страниц; категории, таксономии; модели социальных сетей
- Естественная работа - данные, структура, запросы, индексы

Просто скорость тоже важна



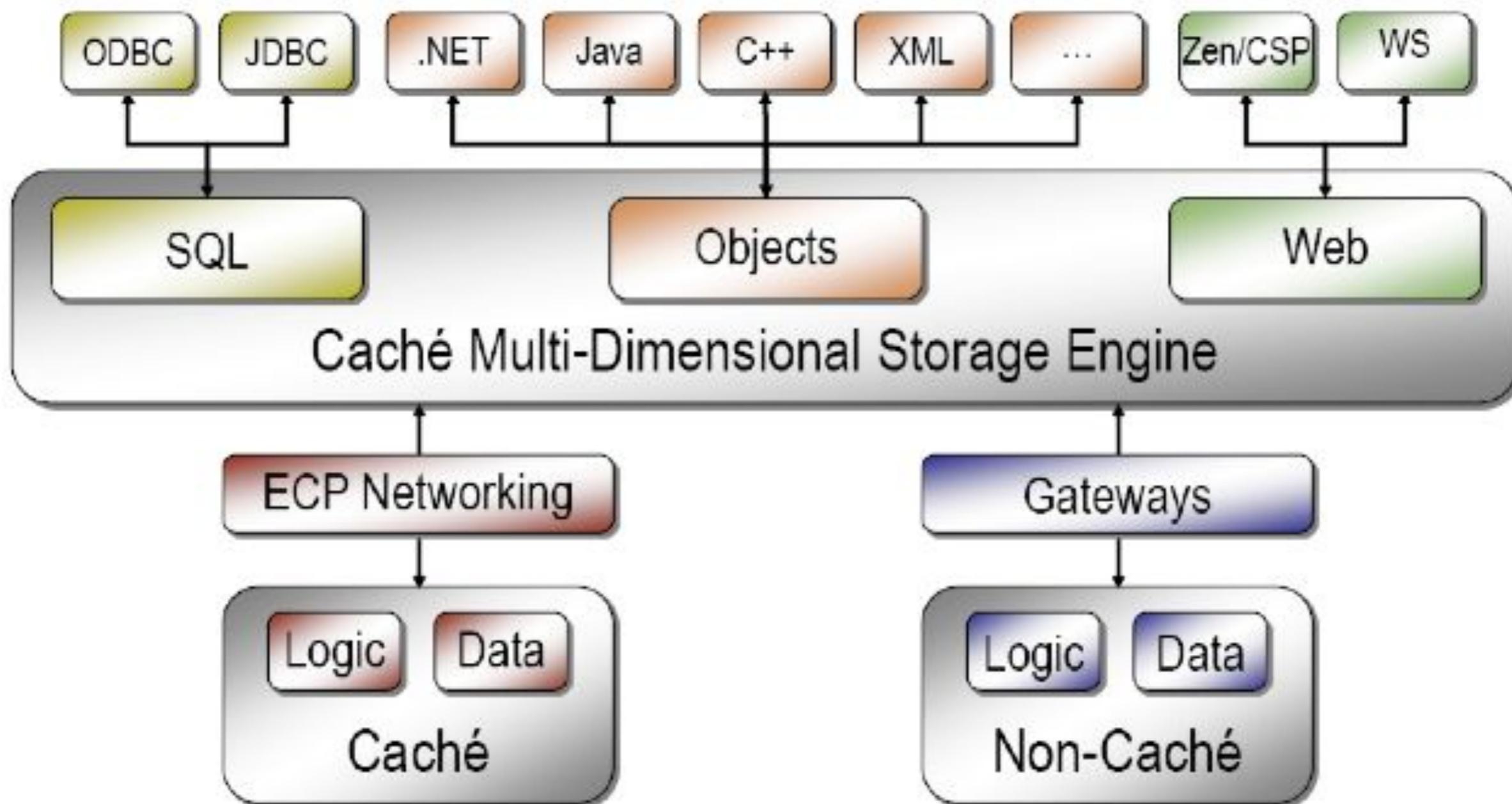
- сохранение 10 000 узлов глобала на этом ноутбуке занимает порядка 0.02 секунды, чтение - порядка 0.01 секунды
- задачи обработки биржевых данных
 - 30 000 - 40 000 ticks / sec - low cost Linux сервера
 - более 80 000 ticks / sec - high end Linux сервера
 - “Игрок” - 6 000 FIX / sec
- скорость установки - 15 минут
- скорость настройки кластера - полчаса вместе с тестированием

Доступ к Caché



- ТСП сервер сохраняющий данные в глобал - 10 строк
- Доступ через HTTP
- Поддержка XML, SOAP, JSON
- Cache Server Pages, Zen Ajax framework
- CSP Gateway
- Механизм проекций: java, c++, .net, python, perl, COM

Технологии InterSystems Caché



Cache как платформа инфраструктурных сервисов Web 2.0



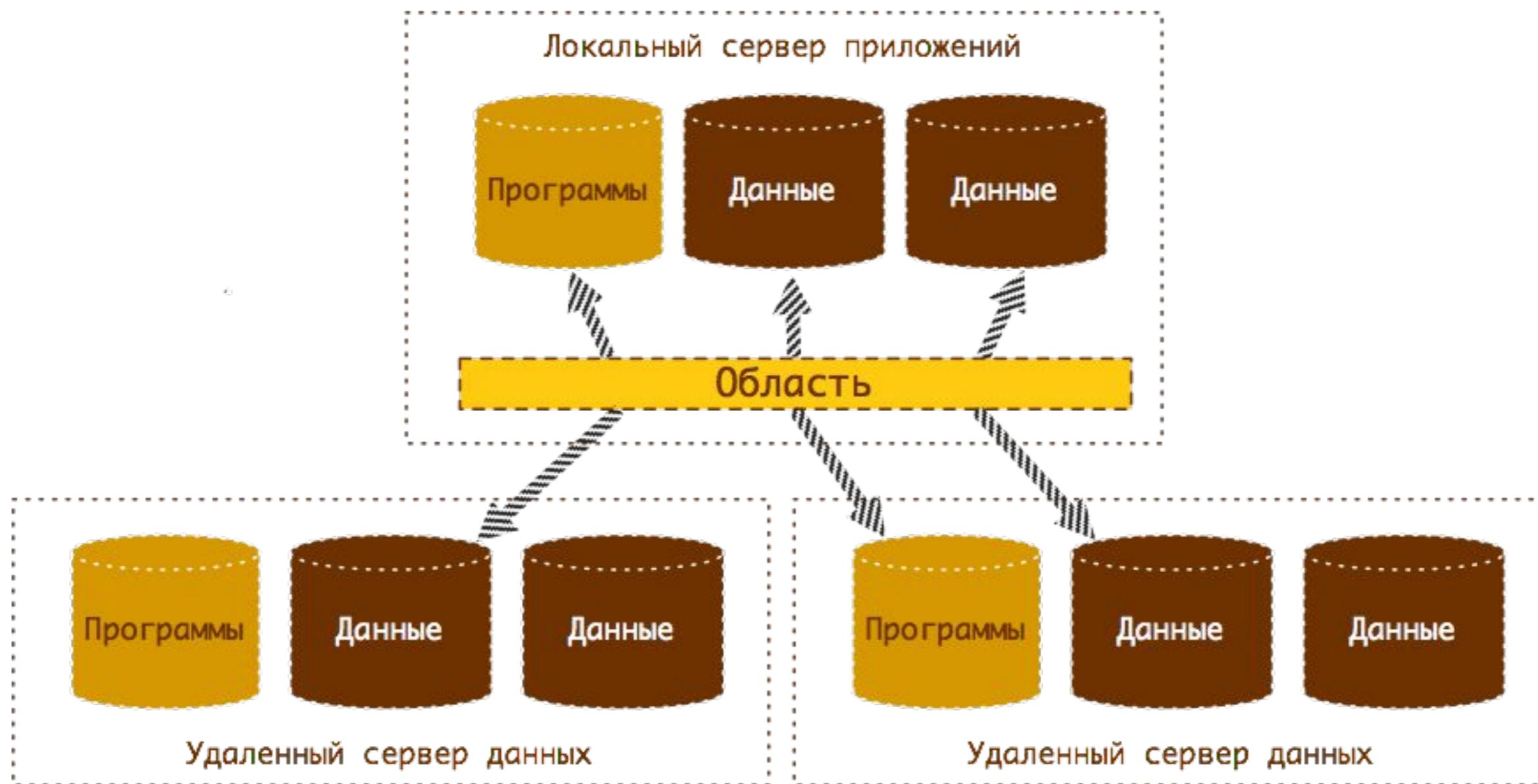
- общедоступные специализированные сервисы, которые позволяют работать с определенными типами данных как приложениям, так и пользователям
 - Flickr, Google Maps, YouTube
- принцип Database as a Service
 - скорость развертывания, zero cost сопровождение, быстрый выход из игры
 - Примеры
 - сервисы Amazon SimpleDB и Amazon S3. Достоинство - нереляционная модель
 - проект M/DB - полный аналог SimpleDB, превышающий Amazon SimpleDB по ряду параметров
 - M/DB успешно функционирует на InterSystems Cache

Построение распределенных баз данных на Caché. Принципы.



- Легкость конфигурации достигается с помощью использования принципа отделения логического доступа к программам и данным от физического их расположения
- Особенность Caché - отсутствие различия между функционированием сервера Caché как сервера приложений и сервером данных
- Subscript Level Mapping - распределение записей по разным базам данных в зависимости от значения полей записи
 - партиционирование данных
 - сохранять данные в зависимости от контекста распределенно
- Enterprise Cache Protocol

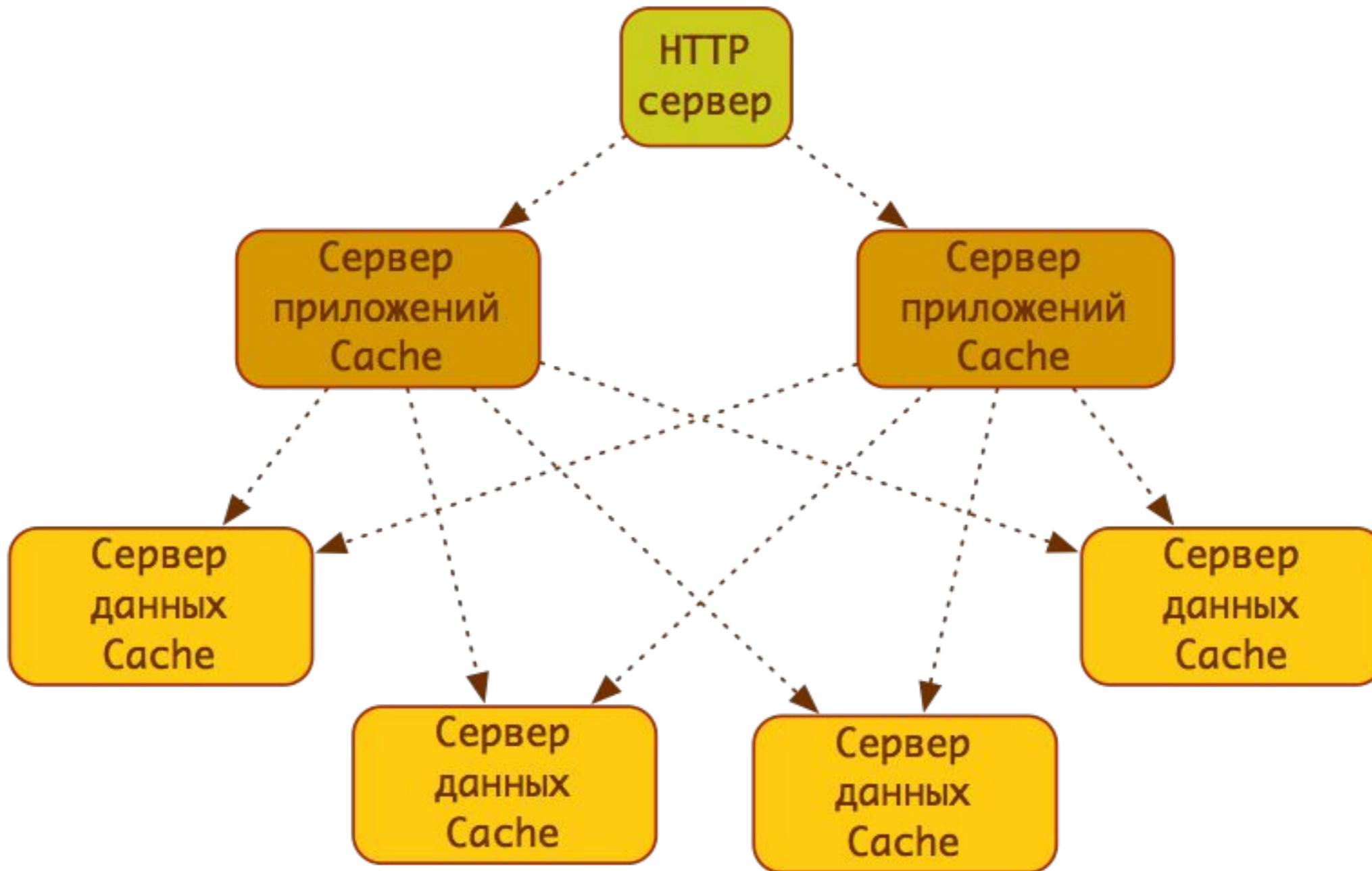
Абстракция доступа



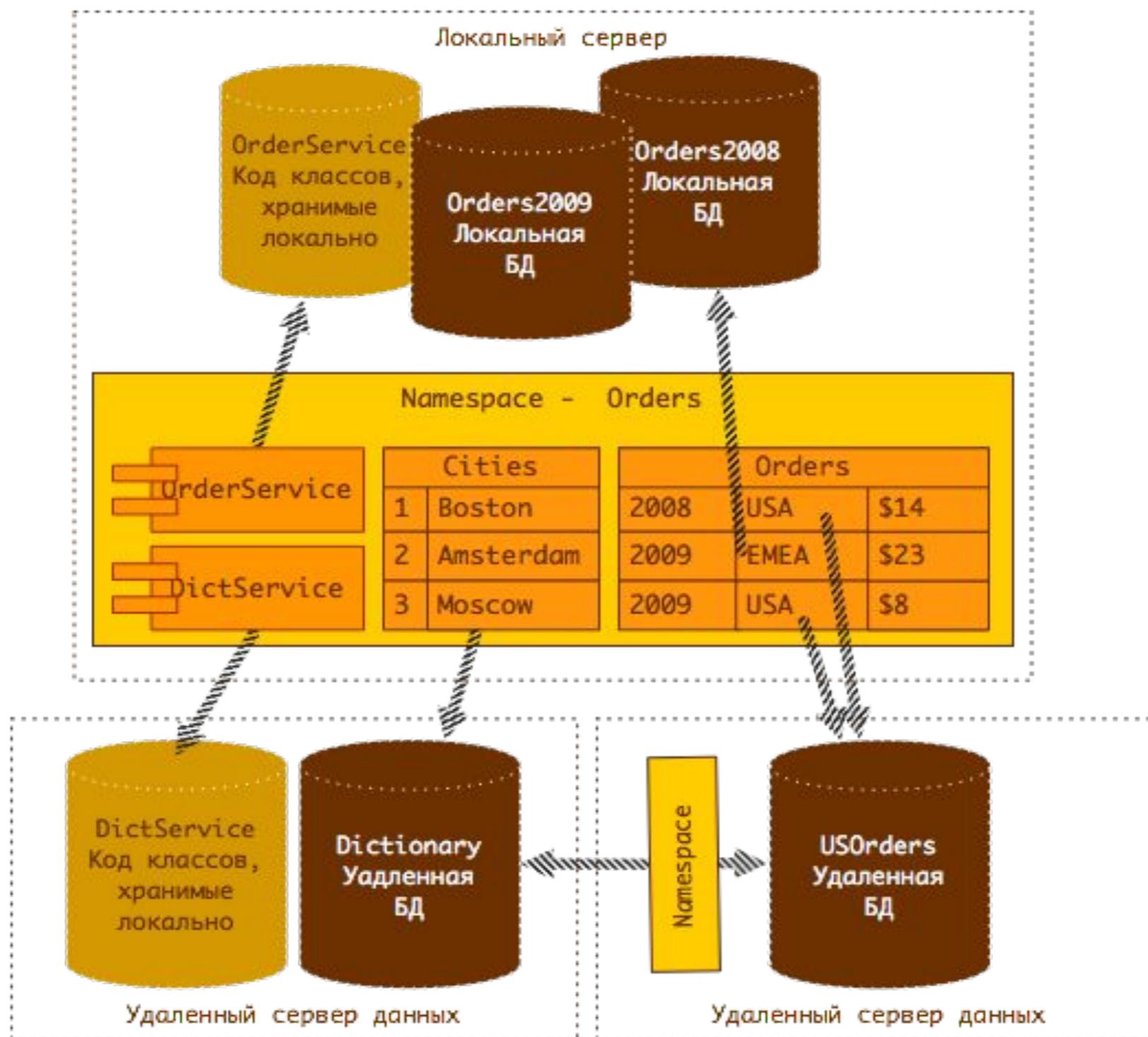


- ECP позволяет конструировать различные по архитектуре и топологии кластера:
 - Распределения вычислительной нагрузки среди нескольких серверов приложений Caché.
 - Преодоления ограничений по размерам оперативной памяти одного сервера при работе нескольких приложений с большим объемом кэшируемых в ОЗУ данных.
 - Распределения хранимых данных между несколькими серверами данных
 - Создания географически распределенной базы данных с возможностью распространения изменений в реальном времени
 - Создания отказоустойчивого решения с дублированием серверных ресурсов
- ECP - гетерогенная технология. Увеличение мощностей по требованию
- На практике, при использовании кластеров Caché обеспечивается одновременная работа нескольких тысяч конкурентных пользователей с кластером

Enterprise Cache Protocol



ЕСР и SLM вместе





- Общая тенденция развития веб приложений сравнимы с тем, к чему стремиться IT архитектура предприятий.
 - приложения, использующие общие инфраструктурные сервисы,
 - доступные по требованию в любое время и с любого места
 - практики SOA
- В случае предприятия SOA часто сводится к построению оптимальной инфраструктуры

Caché и Enterprise 2.0



- Caché в инфраструктуре предприятия
 - специализированный модуль инфраструктуры
 - повышенные требования по производительности
 - необычность и сложность задачи
 - альтернатива Java EE серверам
 - архитектурно - сервер приложений
 - реализованы технологии интеграции
 - все рядом и проще - быстрое средство разработки

Work with InterSystems.

Not separate systems.



Спасибо, пишите

oleg.olenin@intersystems.com

INTERSYSTEMS