Любые данные, которые хранятся в базе данных Oracle, просто обязаны существовать в каком-либо табличном пространстве. Под табличным пространством (tablespace) понимают логическую структуру, то есть вы не сможете попросить ОС показать вам табличное пространство Oracle.

При этом каждое табличное пространство включает в себя физические структуры, называемые файлами данных (data files). Одно табличное пространство Oracle способно содержать один либо несколько файлов данных, в то время как каждый файл данных может принадлежать лишь одному tablespace. Создавая таблицу, мы можем указать, в какое именно табличное пространство мы её поместим — Oracle находит для неё место в каком-нибудь из файлов данных, которые составляют указанное табличное пространство.

Создавая новую таблицу, мы можем поместить её в табличное пространство DATA1 либо DATA2. Таким образом, физически наша таблица окажется в одном из файлов данных, которые составляют указанное табличное пространство.

**Файлы базы данных Oracle**

База данных Oracle может включать в себя физические файлы 3-х основных типов:

• control files — управляющие файлы;

• data files — файлы данных;

• redo log files — журнальные файлы либо журналы.

Посмотрим на отношения между ними:

В управляющих файлах содержится информация о местонахождении других физических файлов, которые составляют базу данных Oracle, — речь идёт о файлах данных и журналов. Также там хранится важная информация о содержимом и состоянии БД Oracle. Что это за информация: • имя базы данных Oracle; • время создания БД; • имена и местонахождение журнальных файлов и файлов данных; • информация о табличных пространствах; • информация об архивных журналах; • история журналов, порядковый номер текущего журнала; • информация о файлах данных в автономном режиме; • информация о резервных копиях, контрольных точках, копиях файлов данных.

При этом функция управляющих файлов не ограничивается хранением важной информации, нужной при запуске экземпляра, — полезны они и в процессе удалении БД Oracle. К примеру, уже с версии Oracle Database 10g можно посредством команды DROP DATABASE удалить все файлы, которые перечислены в управляющем файле БД, включая сам управляющий файл.

### Механизм восстановления данных

В СУБД Oracle можно включить архивацию вышеописанных оперативных журнальных файлов, и все изменения будут архивироваться. Таким образом при потере любого диска с блоками данных мы можем восстановить их на любой момент времени, включая момент прямо перед падением, накатив на последние архивные журнальные файлы текущий оперативный журнал.

### Что позволяет БД Oracle работать так быстро?

Когда вы меняете данные в БД, то ваши изменения сначала идут в кэш, а потом асинхронно в нескольких потоках (число можно сконфигурировать) пишутся на диск. Синхронно же пишется специальных лог (оперативный журнальный файл), чтобы была возможность восстановить данные после сбоя, если они еще не успели с кэша сброситься на диск. Данный подход позволяет выиграть в скорости, так как в этом случае на диск все пишется последовательно в один файл, причем можно настроить так, чтобы писалось параллельно на два или больше дисков, тем самым увеличивая надежность защиты от потери изменений. Описанных файлов должно быть несколько, и они используются по кругу: как только все данные защищенные одним из лог файлов были записаны фоновым процессом в блоки данных на диск, то данный лог файл может быть переиспользован. Таким образом в какой-то мере это позволяет еще и сэкономить, имея ультрабыстрые диски небольшого размера только для небольших журнальных файлов используемых по кругу.

### Пустые строки В оракл есть одна очень интересная особенность, от которой они теперь уже никогда не смогут избавиться. Дело в том, что если вы кладете в БД пустую строку, то она сохраниться как NULL. Таким образом при последующем чтении вы никогда не получите пустой строки, а только NULL. Имейте так же в виду, что по этой же причине пустые строки не попадают в индекс, так что если вы будете делать запросы, план выполнения которых, будет использовать индекс, то ваше пустые (вернее NULL) строки вы никогда не получите