<https://habrahabr.ru/post/111207/>

#### **1. CT (Change Tracking).**

Зачастую путают с CDC (Change Data Capture). Но эти инструменты различны как в назначении, так и в реализации. CT предназначен для отслеживания фактов изменений (в каких строках, какие данные были изменены (CRUD)), в то время как CDC хранит историю изменений (все версии строк, в том числе те, которые были удалены). Что касается реализации, CDC основан на чтении журнала транзакций (асинхронен), в то время как CT работает синхронно.

Для каждой таблицы, для которой включено отслеживание изменений, создается системная таблица, в которой хранился ID измененной строки, битовая маска для идентификации измененных колонок, тип операции.

Для включения CT нужно активировать его на уровне БД и для конкретной таблицы:

ALTER DATABASE ChangeTracking SET change\_tracking = ON<br/>

(change\_retention = 10 minutes, auto\_cleanup = ON) <br/>

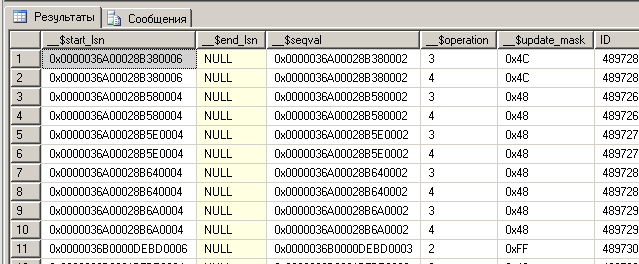
<br/>

ALTER TABLE Orders enable change\_tracking WITH (track\_columns\_updated = ON)

Более детально (описание параметров, примеры использования и детальная информация) в отличной [статье](http://blogs.msdn.com/b/alexejs/archive/2009/08/09/change-tracking.aspx).

#### **2. CDC (Change Data Capture)**

Средство для отслеживания измененных данных. Основными отличиями от CT являются асинхронная реализация (как писалось выше) и хранение всех версий измененных (CRUD) данных. Для хранения измененных данных CDC использует системные таблицы в схеме cdc. Для каждой таблицы, для которой активирован CDC, создается таблица с названием по типу cdc.dbo\_Orders\_CT (для таблицы dbo.Orders).



* \*\_lsn – (log sequence number ) — своего рода идентификаторы транзакции в логе.
* $operation — тип операции (1 — delete, 2 — insert, 3 — update (версия строки до обновления), 4 — update (версия строки после обновления).
* $update\_mask — битовая маска, указывающая на изменившиеся ячейки в строке.
* ID и все остальные справа — ячейки соотв. структуре таблицы, изменения в которой отслеживаются.

Для активации CDC Вам нужно активировать его на уровне БД для конкретной таблицы:

EXEC sys.sp\_cdc\_enable\_db<br/>

<br/>

EXEC sys.sp\_cdc\_enable\_table<br/>

@source\_schema = N'dbo',<br/>

@source\_name = N'Orders',<br/>

@role\_name = N'cdc',<br/>

@capture\_instance = N'dbo\_Orders',<br/>

@supports\_net\_changes = 1,<br/>

@index\_name = 'id\_idx',<br/>

@captured\_column\_list = null,<br/>

@FILEGROUP\_NAME = null;

* @source\_schema — схема, которой принадлежит таблица, для которой активируем CDC
* @source\_name — название таблицы, для которой активируем CDC
* @role\_name — имя роли, которая будет иметь право на просмотр изменений (при отсутствии — создается автоматически)
* @capture\_instance — соответствует части названия, которое будет выдано соответстующей системной таблице
* @supports\_net\_changes — поддержка возможности отображения результирующего набора изменений (только последние версии данных). Для этого необходимо наличие уникального индекса
* @index\_name — собственно, название уникального индекса
* @captured\_column\_list — список полей, для которых будет активировано отслеживание изменений. По умолчанию — все.
* @filegroup\_name — файловая группа, в которой будут размещена системная таблица

С чисто практической точки зрения, значительный минус CDC это то, что невозможно зафиксировать автора изменений. Конечно, никто не мешает добавить столбец в системную таблицу cdc.dbo\_Orders\_CT с дефолтным значением suser\_sname() (в моей практике это работает), но подобные манипуляции с системными таблицами — не лучший способ построения отказоустойчивой системы.

Примеры запросов к сохраненным данным и развернутое описание в [статье](http://blogs.msdn.com/b/alexejs/archive/2009/08/07/cdc.aspx).