Unique, Pseudo Random Numbers

Generate without using RAM, Disk, CPU, I/O, or even time ☺

Version: 1.2 (20191211)

C:\> whoami

- Founder of Sql Quantum Lift:
  - **SQL#** (SQLsharp) : SQLCLR library of functions
  - OmniExec : Multi-threaded, multi-server & DB query tool
- Blog: Sql Quantum Leap
- Areas of interest / concentration:
  - Module Signing, Collations & Encodings, SQLCLR
- Articles:
  - SQL Server Central (incl. Stairway to SQLCLR series)
  - Simple-Talk
- Working in IT and with databases since 1996:
  - SQL Server (since 2002), SQLCLR (since 2006), specializing in Collation & Module Signing (since 2014)
- Variety of Roles, OSes, Languages, and DBs

Agenda

- Why?
- What is Random?
- Randomization Options
- Unique-“Random” Options
- The Maths
- The Code
- The Examples

Why?

- Hide “ID” values from external uses
- Cryptography:
  Mathematics of Cryptography
  (http://highered.mheducation.com/sites/dl/free/0072870222/385983/ch02.pdf)
- Generating test data
- “Shuffle” finite data set

- Cool party trick / impress others ?
- ????

What is Random?

- Unpredictable order?
- Unique?
- Both?
  - If unique is required, need to give up “pure” unpredictability
  - Similar to NoSQL vs ACID-compliant SQL

Randomization Options

- RAND()
- RAND(*seed*)
- NEWID()
- CRYPT_GEN_RANDOM(*length*)
- CRYPT_GEN_RANDOM(*length, seed*)

(example code)

Randomizing Tests

UniquePseudoRandomNumbers_01_RandomizingTests.sql

(http://pastebin.com/vuK1Aiiv)

Unique-“Random” Options

- Pre-generate list
  - Can have no pattern
  - Requires lots of memory and/or disk
  - Requires lots of I/O
- Pseudo-random
  - Uses very little RAM, Disk, or I/O
  - Pattern, but not discernible

The Maths

- Modular multiplicative inverse
  (https://en.wikipedia.org/wiki/Modular_multiplicative_inverse)

$$a\,x \equiv 1 \pmod{m}.$$

- Bézout's identity
  (https://en.wikipedia.org/wiki/B%C3%A9zout%27s_identity)

$$ax + by = d.$$

- Extended Euclidean algorithm
  (https://en.wikipedia.org/wiki/Extended_Euclidean_algorithm)

$$ax + by = \gcd(a, b).$$

- Greatest common divisor
  (https://en.wikipedia.org/wiki/Greatest_common_divisor)

Sql Quantum Leap

(example code)

Create Objects

UniquePseudoRandomNumbers_02_CreateObjects.sql

(http://pastebin.com/edit/n6k8DeyC)

How to Use

- Pick upper limit (this will be the Modulo)
- Try GCD until = 1 (this will be the "base" value)
- Find MMI with Modulo and Base value
- Encode with Modulo and base value
- If / when necessary, decode with Modulo and MMI

(example code)

Multiplicative Inverse Tests

UniquePseudoRandomNumbers_03_MultiplicativeInverseTests.sql

(http://pastebin.com/7u2NzYtd)

Also See

- Generate different random time in the given interval
  (https://stackoverflow.com/a/27826049/577765)
- Generating a random, non-repeating sequence of all integers in .NET
  (https://stackoverflow.com/a/35103941/577765)

- Khan Academy: Modular inverses
  (https://www.khanacademy.org/computing/computer-science/cryptography/modarithmetic/a/modular-inverses)
- Algorithm in many languages: http://rosettacode.org/wiki/Modular_inverse
- Online Calculator: http://planetcalc.com/3311/

- Eric Lippert: A practical use of multiplicative inverses
  (https://ericlippert.com/2013/11/14/a-practical-use-of-multiplicative-inverses/)
- generate seemingly random unique numeric ID in SQL Server
  (https://stackoverflow.com/q/26967215/577765)

Hiding in Plain Sight

- Company:
  - https://SqlQuantumLift.com/
- Blog:
  - https://SqlQuantumLeap.com/
- Articles:
  - https://www.SqlServerCentral.com/author/solomon-rutzky
  - https://www.SqlServerCentral.com/stairways/stairway-to-sqlclr (Stairway to SQLCLR)
  - https://www.simple-talk.com/author/solomon-rutzky/
- SQLsharp.com
  - https://SQLsharp.com/
- StackOverflow.com & DBA.StackExchange.com
  - https://StackExchange.com/users/281451/solomon-rutzky
- LinkedIn
  - http://www.LinkedIn.com/in/srutzky/
- Email:
  - SRutzky@SqlQuantumLift.com