Random number generators minimum technical requirements

Version 1.4.1

For further information, please contact the Office of Liquor and Gaming Regulation on 13 QGOV (13 74 68) or visit www.business.qld.gov.au/industry/liquor-gaming

Contents

Introduction

Policy

All Random Number Generators must be submitted to OLGR for evaluation and approval under the following Legislation:

- Casino Control Act
- Charitable and Non Profit Gaming Act
- Gaming Machine Act
- Keno Act
- Lotteries Act
- Wagering Act

Purpose

The purpose of this document is to:

- Advise the industry of OLGR's minimum technical requirements for Random Number Generators.
- Ensure requirements are consistently applied.
- Achieve a high standard of integrity of random number generators used in gaming in Queensland.
- Ensure that Random Number Generators used in Queensland are sufficiently random for their intended purpose.
- Maintain the integrity of gaming in Queensland by ensuring that all Random Number Generators in use are secure for their purpose and that their outcomes are sufficiently hard to predict under all conditions.

Scope

This document is applicable to all gaming providers in Queensland. These requirements apply to the technical evaluation of Random Number Generators, submitted for evaluation after the date of publication of this document, by OLGR for use in gambling in Queensland.

1 Definitions / Abbreviations

Cycle

1. With respect to a PRNG, “cycle” refers to the number of results generated by a PRNG before it starts repeating the previous sequence of raw results.

2. With respect to a game or application, it refers to the total number of theoretical possible outcomes in the game or application.

Cycling

“Cycling” refers to the process of continuously obtaining a new raw value from a PRNG and throwing it away without making any use of the value. This is typically done to help prevent the deduction of the current state of a PRNG.

Entropy Gathering

Entropy is needed to seed an RNG and is also used to compliment the state of an operating RNG. Entropy is the randomness which occurs as the result of a process. Entropy for instance can be produced by timing the time between entry key strokes.

Mapping

Refers to the process of selecting an outcome using the result from an RNG.

Pseudo Random Number Generator (PRNG)

A type of RNG implemented in software or firmware which generates predictable sequence of numbers with the intended property of statistical randomness. PRNGs utilise what is referred to as a “deterministic algorithm”, in that future outcomes are directly dependent on previous outcomes.

Random Number Generator (RNG)

Refers to any item of hardware or software which can generate random numbers with the intended property of statistical randomness. An RNG may be composed of several other RNGs. Accordingly, the term RNG can also be used to refer to a specific “underlying” RNG within an overall RNG. The operation of RNGs can be split into the areas as follows (NB. not all areas are applicable depending on the type of RNG): Seeding, Underlying RNGs, Scaling, Mapping, Cycling and Entropy Gathering.

Range

Refers to the size of the values, and the lowest and highest value, required to determine the outcome.

Raw Results

The term; “Raw Results” refers to results from an RNG before any scaling or mapping takes place.

Seed

Refers to the initial state information typically required for PRNGs and in some cases Mechanical RNGs as well.

Seeding
Refers to the process used to determine the initial state of the RNG. Typically, seeding is applicable to PRNGs and some Mechanical based RNGs only.

Scaling
Refers to the methodology used to convert the raw results of an underlying RNG into the required range of output values. An example of this would be where a random number from 1 to 100 is required and the RNG used produces random numbers between 1 and 2^{32}. Scaling refers to the process used to convert from the potentially large numbers produced by the RNG into the smaller number range required.

Underlying RNG
The low level RNG used to generate raw results.

XOR
The term; "XOR" refers to the process of combining two different raw results in binary form where if the same two values are combined it results in a 0 and if the values are different it results in a 1.

2 Categories of Random Number Generators

2.1 Pseudo Random Number Generators (PRNGs)

Refer to the Definitions / Abbreviations section.

2.2 Mechanical RNGs

A mechanical device used to generate random results excluding 'Hardware RNGs' defined below. Examples of Mechanical Random Number Generators are: roulette wheels, dice and ball draw machines.

2.3 Hardware RNGs

Refers to any RNG based on an electronic circuit or Integrated Circuit (IC). An Integrated Circuit is a silicon chip containing an electric circuit made up of components such as transistors, diodes, resistors and capacitors. Integrated circuits are smaller, faster and more efficient than the individual circuits used in older computers.

3 RNG Submissions

3.1 A submission under these requirements must comply with the latest version of the OLGR "Submissions Requirements" document available for download from the OLGR website.

3.2 Submit a fully functional RNG device. If this is not possible detail the reasons why it is not possible and have a fully functional RNG device available for access by OLGR staff.

3.3 In the submission letter, designate the overall submitted RNG with a name and version number for identification purposes.

3.4 Submit all scaling/mapping algorithms to be used in conjunction with the submitted RNG. Each of these algorithms must be provided with a unique name or designation and version number.

3.5 Supporting references/academic reviews of the PRNG/scaling algorithms are also desirable if available.

3.6 Clearly state in the submission letter the RNG's intended use. This must also be in regard to the frequency of calling, the required result ranges, the expected application cycle requirements, maximum prize size, number of prizes drawn and frequency of draws. (NB. The RNGs will be evaluated and approved for use with respect to these items.)

3.7 Supply all applicable source code for the overall RNG including scaling and mapping algorithms. Where possible the source code must be able to be compiled and some sample output data provided to verify the compiled source code against. A method to verify the provided source code is used in respect to the live system may be mandated at the OLGR's discretion and depends on overall risk.

3.8 Supply details and results of all tests performed on the RNG prior to submission. Any test information provided to OLGR can reduce the time and cost of the RNG evaluation.

3.9 Supply sample sets of results from all the RNGs within the overall RNG for both before and after scaling/mapping (if applicable). These results must be provided in a text file and for hardware & PRNGs. The files must contain at least 500,000 results each.

3.10 Mechanical based RNGs must be submitted with all custom or non-commercially available equipment/gear/harnesses required to fully test them.

3.11 Mechanical based RNGs must be submitted with all equipment required in order to commission, decommission and operate the device.

3.12 Submit the methodology, procedure, time interval and tests performed for any system monitoring which occurs to ensure continuing health of the RNG.

3.13 Provide literature which supports the ability of the RNG to produce the required level of random results. If possible, provide links to reviews published on the internet that detail capabilities of or testing done on the RNG.

4 General

The following requirements apply to all RNG types except where otherwise stated.

4.1 Randomness

RNGs must produce results which can be proven to be:

a) statistically independent
b) uniformly distributed over the range of results
c) unpredictable
d) pass specified statistical tests (see below)

4.2 RNG Statistical Tests that RNGs must pass (not all tests are applicable in all cases, it is at OLGR's discretion which tests are applied):

a) Equi-Distribution (Frequency) Test
This test is used to check that the frequencies of the results are approximately uniformly distributed. If the frequencies are too close to the expected frequencies they are too uniform and therefore are not random. If the frequencies are too far from the expected frequencies the results are exhibiting some bias and therefore are not random.

b) Gap Test
This test is used to check that there is a random interval (gap) between a result and the same result re-occurring.

c) Poker Test
This test examines groups of 5 results at a time looking for certain patterns that would constitute different hands in a poker game. The chi-squared test is used to determine if the frequencies of these patterns occurring are approximately the same as the expected frequencies.

d) Coupon Collector's Test
This test examines the interval required for each possible result in the range to occur.

e) Permutation Test
This test checks for dependencies between results. It examines how often one result occurs in conjunction with every other result.

f) Run Test
This test checks that patterns of results do not reoccur more frequently then expected for a random process. This test also checks for dependencies between results. It examines the lengths of each subsequence of monotonically increasing results ("runs up") and decreasing results ("runs down").

g) Spectral Test
This test uses a visual mapping of results to detect any patterns that may indicate that the results were not randomly generated.

h) Serial Correlation Test
This test checks that consecutive results are independent.

i) Tests on Sub-Sequences
This test checks that smaller subsets of the results display the characteristics of randomness.

These tests are applied to individual results and where bets can be placed on multiple grouped results simultaneously then on all possible different grouped results. This would apply in cases such as roulette wheels where players can bet on odds / evens, large / small, street, corner etc. This checks that any small biases that may not be significant for single results are still not significant when grouped results are considered.

4.3 The Chi-Squared Test

- The chi-squared test is the statistical test used to determine whether the RNG passes each of the above tests.
- Chi-squared test on the results should be within the expected range using a 95% confidence interval.
- Each test should be performed on a minimum of 100,000 results for PRNGs and Hardware RNGs. Smaller sets of results can be tested for Mechanical RNGs.

4.4 The state information of an RNG must not be externally visible or accessible. An exception to this requirement may be granted for state information which if made public, is proven that in no way could possibly compromise the security or integrity of the game utilising the RNG. E.g. For ball drawing RNGs the state information is the current instantaneous location of the ball in the machine.

4.5 The cycle of the RNG must be greater than the largest range required by its application by at least a factor of four.

4.6 The range of values produced by the RNG must be adequate to provide sufficient precision and flexibility when setting event outcome probabilities.

4.7 There must be no external mechanism (switches, jumpers, etc.) that can affect the outcome of the RNG. For example, there shall be no externally selectable options that alter or affect the RNG.

4.8 RNGs that gather additional entropy over time or operation will be evaluated form the point of view that all sources of additional entropy have failed or are corrupt, unless the source of injected entropy is demonstrated to be failsafe.

(While injecting external sources of entropy into the state of an RNG is strongly encouraged, this is generally not a contributing factor in the evaluation of the overall RNG unless it is a hardware-based RNG that is injecting the entropy.)

4.9 Any RNG submitted must have a minimum cycle of 2^32.

4.10 Any RNG used for the game of Keno or similar cycle game and prize game, must utilise an RNG comprised with a minimum of a PRNG XOR'ed with a hardware based RNG.

4.11 At the application level, random numbers for a game must not be generated (if possible) until all player input choices or bets have been made, i.e. the game must be 'closed' first. There must not be any interval where access to the result could accrue a benefit.

4.12 The RNG must be secure in proportion to its level of responsibility of its application and the size of the prizes awarded as a result of the RNG outcome i.e. the risk to the integrity of gaming and community impact of fraud in relation to the RNG.

5 Hardware RNGs

Requirements specific to Hardware based RNGs.

5.1 Hardware based RNGs must not be utilised as stand-alone RNGs. However, they may be used to compliment (XOR) the state or output of another underlying PRNG. They also are highly recommended as a source for seeding the initial state of an RNG.

6 Mechanical RNGs

Requirements specific to Mechanically based RNGs.

6.1 A Mechanical RNG must not have any taint, bias or patterns without acceptable fail-safes in place. (E.g. rotating canoe rings in roulette wheels). If a product may develop a taint, bias or pattern through wear, the product must be monitored or tested for this on an ongoing basis and a procedure for this must be included in the RNG submission.

6.2 Initial starting positions, if any, must not correlate to output positions. (E.g. ball drawing devices).

6.3 The device must operate correctly if the surface it sits on is not level (if applicable) unless there is an acceptable fail-safe procedure in place to ensure the device remains level during operation.

6.4 If possible Mechanical RNGs must be submitted with the ability to be rigged up to enable the device to be auto-played (no manual intervention) and the results automatically electronically recorded for an indefinite period. If this is not possible, then results from the RNG under conditions specified by OLGR may be required as a part of the evaluation.

6.5 Mechanical RNGs must also be submitted with the following information for evaluation and approval:

- Commissioning / decommissioning procedures,
- Operating procedures,
- Care & preventative maintenance procedures.

The main requirements for the documents are completeness, correctness and that a non-technical person must be able to follow them. These documents will also be evaluated in regards to the likelihood of the procedures preventing or causing bias or taint to the mechanical device and detecting it if it occurs.

6.6 If the device is prone to mechanical wear, especially wear that can adversely affect the randomness of the results, then provide documentation or manuals which include the maintenance procedures for the device. The preventive maintenance procedures provided must be detail best practice procedures to reduce the risk of wear occurring and detect any wear which negatively impacts the randomness of the device results.

6.7 There may be a requirement imposed to monitor the results of the device to ensure no degradation occurs during operation.

6.8 Results from the RNG must be clear and not open to confusion.

6.9 If possible, no mapping of results should be used. Any mappings used must be obvious, unambiguous and clearly visible to players e.g. odd and even results in Roulette.

6.10 Any ability to tamper with the device in such a way as to affect the results produced should be minimised and controlled. Any attempted tampering must be clearly visible to a casual observer or easily tested for.

6.11 Any damage to the device that affects the randomness of the results should, if possible, produce no result and clearly display an error state, or visible change to the equipment or the draw process, or be able to be tested for.

6.12 Packaging for transportation of the Mechanical RNG device must be:

- Re-useable
- Considered sufficient by the CEO, OLGR to ensure the device's randomness will not be adversely affected during normal transportation of the device.

6.13 In the case of a mechanical RNG there may also be a maximum lifetime limit imposed or a maximum cycle.

7 Seeding

Some RNGs such as PRNGs require seeding before use. These requirements apply to RNGs that require seeding.

7.1 The seeding process must be a random process in itself, subject to all requirements in this document concerning randomness.

7.2 The RNG must gather sufficient random entropy before seeding and the entropy gathered before seeding must be at least as random as results required during normal operation. Gathering random entropy for seeding from hardware based RNG is highly recommended.

7.3 The seeding process must not be visible externally.

7.4 The method of seed generation must ensure that when a duplicate of the RNG is used in multiple devices, it is highly improbable that the same initial sequence of random numbers is used in more then one device.

7.5 Seeding must be kept to a minimum, such as seeding every power up of a PRNG and will only be approved under special circumstances and with a strong seeding methodology.

7.6 Where possible, the method for re-seeding upon power up should be a function of all or part of the state of the RNG just prior to power down.

7.7 The RNG initial state must be seeded from at least two reasonable sources of random entropy. For example, a real time clock, hardware RNG, time between two non-deterministic events such as button presses on the device and entropy gathered from the operating system such as clock drift, network traffic and so on.

8 Scaling Algorithms

PRNGs and Hardware RNGs typically require a scaling algorithm in order to take a raw value from the underlying RNG and convert it into the desired range required by the application. The requirements in this section apply to any scaling algorithms utilised in an RNG.

8.1 The scaling algorithms must achieve their intended results with no bias or an insignificant amount of bias.

8.2 Each of the possible outcomes of the scaling algorithm must have the correct desired probabilities as per the mathematical model of the game or application.

8.3 Scaled results must pass all tests for randomness as well as the raw RNG results.

9 Cycling

A PRNG or deterministic RNG must be "cycled" periodically when idle. In this regard these requirements apply to PRNG and deterministic RNGs.

9.1 The minimum acceptable cycle frequency depends on the application, equipment and frequency of results being used, but 200ms is generally considered sufficient.

9.2 Cycling must not be externally visible.

9.3 The cycling of an RNG must not be predictable, i.e. cycling which always occurs the same number of times between every application call of the RNG is not considered acceptable.

9.4 Games in which there is the chance of a very low probability outcome (e.g. a jackpot win, a 1-2-3-4-5 sequence of numbers drawn etc), where the game outcome is represented by only a small number of RNG values, **must** include cycling of the RNG to prevent the next outcomes being predictable.

10 RNG Monitoring

10.1 The randomness of the RNG must be monitored under the following circumstances:

a) The game of keno;
b) Games with a long lifetime with frequent usage and large prizes;
c) Hardware RNGs; and
d) Other games as specified by OLGR.

10.2 Monitoring should detect whether the RNG is not performing acceptably, random results are no longer being obtained or the Hardware RNG (if applicable) has failed.

Two types of randomness tests are appropriate:

a) Small sample – Testing of small samples of results at frequent time intervals allows for the rapid detection of extreme results, which would occur as a result of sudden, complete or near complete failure of the RNG.
b) Large sample – Testing of large sets of results less frequently validates the correct performance of the RNG. This type of test should be designed to detect programming

errors in the RNG and biases that may result from wear or are a product of the original RNG but were not detected.

11 Charitable and Non Profit RNGs

RNGs submitted under the Charitable and Non Profit Act may be approved with the following exemptions from the requirements:

11.1 If a PRNG is used to generate a small number of results only, cycling of the PRNG is not required.

11.2 The hardware clock may be used as the only seed value of the RNG, if the clock is available in units of milliseconds or better, where the result is not easy to predict or control using this information.

11.3 To use an RNG which is part of another commercially available product, where the source code is not available for submission to OLGR, under the following conditions:

a) The RNG algorithm that the commercially available product implements, is known.
b) The commercially available product is a well known product with a demonstrated track record.

12 Revision History

Version	Changes	QIR	Who	Release Date	Incept Date
1.2	Release for industry comment	273	CH & RL	14/03/2006	
1.3	Minor update to incorporate feedback from industry	273	CH	15/06/2006	
1.4	Updated to new DEEDI report document template		YL	20/08/2010	
1.4.1	Updated to new DJAG report document template		JG	19/04/2016	