

# Package ‘ErlangC’

December 5, 2024

**Type** Package

**Title** Solve Erlang-C Model

**Version** 0.1.0

**Maintainer** Damonsoul <chenmaowei96@gmail.com>

**Description** Provides a set of functions to solve Erlang-C model. The Erlang C formula was invented by the Danish Mathematician A.K. Erlang and is used to calculate the number of advisors and the service level.

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**Imports** bsicons, bslib, DT, gmp, lubridate, shiny, tidyR

**RoxygenNote** 7.3.2

**Depends** R (>= 2.10)

**Suggests** testthat (>= 3.0.0)

**Config/testthat/edition** 3

**NeedsCompilation** no

**Author** Damonsoul [aut, cre]

**Repository** CRAN

**Date/Publication** 2024-12-05 18:30:15 UTC

## Contents

calculate_agents . . . . .	2
erlang_c . . . . .	3
erlang_c_app . . . . .	4
translations . . . . .	4

<b>Index</b>	<b>5</b>
--------------	----------

---

calculate_agents	<i>Calculate Required Number of Agents This function calculates the required number of agents to achieve a specified service level and occupancy.</i>
------------------	---

---

### Description

Calculate Required Number of Agents This function calculates the required number of agents to achieve a specified service level and occupancy.

### Usage

```
calculate_agents(
    call_count,
    call_period,
    avg_handle_time,
    target_anser_time,
    require_service_level,
    max_occupancy,
    shrinkage,
    max_agents = NULL
)
```

### Arguments

call_count	Numeric. The total number of incoming calls.
call_period	Duration. The time period over which calls are counted.
avg_handle_time	Duration. The average time taken to handle a call.
target_anser_time	Duration. The targeted time to answer a call.
require_service_level	Numeric. The required service level.
max_occupancy	Numeric. The maximum allowed occupancy level .
shrinkage	Numeric. The shrinkage factor to account for non-productive time .
max_agents	Integer. The maximum number of agents allowed.

### Value

A list containing the calculated metrics and the number of agents required.

**Examples**

```

calculate_agents(
  call_count = 100,
  call_period = lubridate::duration(30, "minutes"),
  avg_handle_time = lubridate::duration(180, "seconds"),
  target_anser_time = lubridate::duration(20, "seconds"),
  require_service_level = 0.8,
  max_occupancy = 0.85,
  shrinkage = 0.3,
  max_agents = 200
)

```

erlang\_c

*Erlang C***Description**

Calculate the performance metrics of an Erlang C model with n agents.

**Usage**

```
erlang_c(call_count, call_period, avg_handle_time, target_anser_time, n)
```

**Arguments**

call_count	Numeric. The total number of incoming calls.
call_period	Duration. The time period over which calls are counted.
avg_handle_time	Duration. The average time taken to handle a call.
target_anser_time	Duration. The targeted time to answer a call.
n	Integer. The number of agents.

**Value**

A list containing the calculated metrics.

**Examples**

```

erlang_c(
  call_count = 100,
  call_period = lubridate::duration(30, "minutes"),
  avg_handle_time = lubridate::duration(180, "seconds"),
  target_anser_time = lubridate::duration(20, "seconds"),
  n = 14
)

```

`erlang_c_app`*Shiny App for Erlang C Calculator*

---

**Description**

This function creates a Shiny app for calculating Erlang C metrics.

**Usage**

```
erlang_c_app(language = "en")
```

**Arguments**

`language` Character. The language to use for translations (default: "en").

**Value**

A Shiny app object.

---

`translations`*Translations*

---

**Description**

This dataset contains translations for all the strings used in the app. It is used to create a Shiny string translation interface.

**Usage**

```
translations
```

**Format**

A data frame with variables:

**key** The key of the string to translate

**en** The translation in English

**zh** The translation in Chinese

**Source**

Local

# Index

\* **datasets**

translations, 4

calculate\_agents, 2

erlang\_c, 3

erlang\_c\_app, 4

translations, 4