|  |
| --- |
| **Software Design Specification** |

GLIMMPSE Domain Objects

Create Date: 2/17/2012

Created By: Uttara Sakhadeo and Sarah Kreidler

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Sarah Kriedler | //2012 | Creation of domain objects shared among all the web services. | 1.1.0 |
| Uttara Sakhadeo | 5/3/2012 | Addition of wrapper classes as a work around for JSON. | 1.2.0 |

Table of Contents

[**Software Design Specification** 1](#_Toc324843000)

[1 Introduction 5](#_Toc324843001)

[1.1 Purpose of this document 5](#_Toc324843002)

[1.2 Scope of the development project 5](#_Toc324843003)

[1.3 Definitions, Acronyms, and Abbreviations 5](#_Toc324843004)

[1.4 References 5](#_Toc324843005)

[1.5 Overview of the document 6](#_Toc324843006)

[2 System Architecture Description 6](#_Toc324843007)

[2.1 Web Services Common Library 6](#_Toc324843008)

[2.2 Integration with Java Web Services 7](#_Toc324843009)

[2.3 Integration with Google Web Toolkit 7](#_Toc324843010)

[3 Module and Component Descriptions 7](#_Toc324843011)

[3.1 Component overview 7](#_Toc324843012)

[3.1.1 The StudyDesign Object 7](#_Toc324843013)

[3.1.2 The StudyDesignList Object 10](#_Toc324843014)

[3.1.3 The Blob2DArray Object 11](#_Toc324843015)

[3.1.4 The NamedMatrix Object 11](#_Toc324843016)

[3.1.5 The NamedMatrixList object 11](#_Toc324843017)

[3.1.6 The NamedMatrixSet object 11](#_Toc324843018)

[3.1.7 The UuidMatrix object 11](#_Toc324843019)

[3.1.8 The UuidMatrixName object 12](#_Toc324843020)

[3.1.9 The BetweenParticipantFactor Object 12](#_Toc324843021)

[3.1.10 The BetweenParticipantFactorList Object 12](#_Toc324843022)

[3.1.11 The Cluster Node Object 13](#_Toc324843023)

[3.1.12 The ClusterNodeList Object 13](#_Toc324843024)

[3.1.13 The Repeated Measures Node Object 14](#_Toc324843025)

[3.1.14 The RepeatedMeasuresNodeList Object 16](#_Toc324843026)

[3.1.15 The Hypothesis Object 16](#_Toc324843027)

[3.1.16 The HypothesisSet Object 18](#_Toc324843028)

[3.1.17 The UuidHypothesis Object 18](#_Toc324843029)

[3.1.18 The UuidHypothesisType Object 18](#_Toc324843030)

[3.1.19 The Covariance Object 18](#_Toc324843031)

[3.1.20 The CovarianceSet Object 19](#_Toc324843032)

[3.1.21 The UuidCovariance Object 19](#_Toc324843033)

[3.1.22 The UuidCovarianceName Object 20](#_Toc324843034)

[3.1.23 The PowerCurveDescription Object 20](#_Toc324843035)

[3.1.24 The UuidPowerCurveDescription Object 22](#_Toc324843036)

[3.1.25 The ConfidenceIntervalDescription Object 22](#_Toc324843037)

[3.1.26 The UuidConfidenceIntervalDescription Object 23](#_Toc324843038)

[3.1.27 The TypeIError Object 23](#_Toc324843039)

[3.1.28 The TypeIErrorList Object 23](#_Toc324843040)

[3.1.29 The BetaScale Object 23](#_Toc324843041)

[3.1.30 The BetaScaleList Object 24](#_Toc324843042)

[3.1.31 The SigmaScale Object 24](#_Toc324843043)

[3.1.32 The SigmaScaleList Object 24](#_Toc324843044)

[3.1.33 The RelativeGroupSize Object 24](#_Toc324843045)

[3.1.34 The RelativeGroupSizeList Object 24](#_Toc324843046)

[3.1.35 The StatisticalTest Object 25](#_Toc324843047)

[3.1.36 The StatisticalTestList Object 25](#_Toc324843048)

[3.1.37 The PowerMethod Object 25](#_Toc324843049)

[3.1.38 The PowerMethodList Object 25](#_Toc324843050)

[3.1.39 The Quantile Object 26](#_Toc324843051)

[3.1.40 The QuantileList Object 26](#_Toc324843052)

[3.1.41 The NominalPower Object 26](#_Toc324843053)

[3.1.42 The NominalPowerList Object 26](#_Toc324843054)

[3.1.43 The ResponseNode Object 26](#_Toc324843055)

[3.1.44 The ResponseList Object 26](#_Toc324843056)

[3.1.45 The SampleSize Object 27](#_Toc324843057)

[3.1.46 The SampleSizeList Object 27](#_Toc324843058)

[3.1.47 The ConfidenceInterval Object 27](#_Toc324843059)

[3.1.48 The FixedRandomMatrix Object 27](#_Toc324843060)

[3.1.49 The PowerResult Object 28](#_Toc324843061)

[3.1.50 The PowerResultList Object 28](#_Toc324843062)

[3.1.51 The StudyNamedMatrixList Object 28](#_Toc324843063)

[3.1.52 The GLMMPower Object 28](#_Toc324843064)

# Introduction

## Purpose of this document

This document describes the domain objects which are shared across software modules within the GLIMMPSE system. The domain objects are represent subcomponents of research study designs, such as variable names, hypotheses, and associated matrices. This document describes the domain objects used in version 2.0.0 of the GLIMMPSE software system.

The domain objects are used to transmit information between the GLIMMPSE user interface and the web services layer. They also provide a convenient format for persisting study design information to a relational database.

## Scope of the development project

The domain layer will be implemented as a new shared library, Web Services Common, for use with Java Web Services and Google Web Toolkit user interfaces.

The existing XML-based communication layer will be replaced with a JSON-RPC layer which transmits the domain objects described in this document. Integration of the new domain objects will impact the following services:

* Study Design Service1
* Matrix Service2
* Power Service3

## Definitions, Acronyms, and Abbreviations

**XML**4- Extensible Markup Language (XML) is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable.

**JSON**5 - JavaScript Object Notation, is a lightweight text-based open standard designed for human-readable data interchange.

**JSON-RPC**6 – A remote procedure call protocol encoded in JSON.

**AJAX** – asynchronous HTTP request. In this context, AJAX requests are issued to update the study design information with the Study Design Service, or to perform a matrix operation.

**Warfile** – web application archive file. This format is used to run web applications under Apache Tomcat.

**Google Web Toolkit**7 – A Google package for creating browser-independent web user interfaces

## References

1. Sakhadeo U. Software Design Specification: Study Design Web Service. 2011.

2. Akula V. Software Design Specification: Matrix Web Service. 2012.

3. Kreidler S. Software Design Specification: Power Web Service. 2012.

4. Anon. Extensible Markup Language (XML) 1.0 (Fifth Edition). Available at: http://www.w3.org/TR/REC-xml/. Accessed February 10, 2012.

5. Anon. JSON. Available at: http://www.json.org/. Accessed February 10, 2012.

6. Anon. specification – JSON-RPC. Available at: http://json-rpc.org/wiki/specification. Accessed February 10, 2012.

7. Anon. Tutorial Overview - Google Web Toolkit - Google Code. Available at: http://code.google.com/webtoolkit/doc/latest/tutorial/. Accessed February 17, 2012.

8. Leach P, Mealling M, Salz R. A Universally Unique Identifier (UUID) URN Namespace. Available at: http://www.ietf.org/rfc/rfc4122.txt. Accessed February 17, 2012.

9. Kreidler S. Software Design Specification: Full Factorial Hypotheses. 2011.

10. Simpson SL, Edwards LJ, Muller KE, Sen PK, Styner MA. A linear exponent AR(1) family of correlation structures. *Stat Med*. 2010;29(17):1825-1838.

11. Taylor DJ, Muller KE. Computing Confidence-Bounds For Power And Sample-Size Of The General Linear Univariate Model. *American Statistician*. 1995;49(1):43-47.

12. Davies RB. Algorithm AS 155: The Distribution of a Linear Combination of Chi-Square Random Variables. *Applied Statistics*. 1980;29(3):323-333.

13. Muller KE, Barton CN. Approximate Power for Repeated-Measures ANOVA Lacking Sphericity. *Journal of the American Statistical Association*. 1989;84(406):549-555.

14. Muller KE, Edwards LJ, Simpson SL, Taylor DJ. Statistical tests with accurate size and power for balanced linear mixed models. *Stat Med*. 2007;26(19):3639-3660.

## Overview of the document

In Section 2, we describe the integration of the domain object layer and Web Services Common library into the GLIMMPSE software system. In Section 3, we detail each domain object.

# System Architecture Description

## Web Services Common Library

The domain object layer will be defined in a Java shared library called Web Services Common. The library also contains utility routines for UUID handling and database interaction which are shared across multiple web services. Classes for each domain object are found in the edu.ucdenver.webservice.common.domain package.

The Web Services Common library is compiled into two jar files:

* edu.ucdenver.bios.webservice.common-*version*-jar
* edu.ucdenver.bios.webservice.common-gwt-*version*-jar

The first is designed for integration with Java Web Services, and the second for Google Web Toolkit projects.

## Integration with Java Web Services

The edu.ucdenver.bios.webservice.common-*version*-jar shared library for Web Services Common can be integrated with Java Web Services. For development, the file should be included in the Java classpath. For deployment, the library should be in the “libs” directory of the warfile.

## Integration with Google Web Toolkit

The edu.ucdenver.bios.webservice.common-gwt-*version*-jar shared library for Web Services Common can be integrated with a Google Web Toolkit project. Unlike a standard jar file, this jar includes both class files and Java source files. The Google Web Toolkit compiler requires the source files to facilitate translation of the classes into JavaScript.

For development, the jar should be included in the Java classpath. In addition, the module must be inherited in the gwt.xml file (see Google documentation for full details7) as follows

Figure 1. Example gwt.xml file including the Web Services Common module

<?xml version="1.0" encoding="UTF-8"?>

<module rename-to='*project*'>

<!-- Inherit the core Web Toolkit stuff. -->

<inherits name='com.google.gwt.user.User'/>

…

<!-- Other module inherits -->

<!-- UC Denver Web Service Common api -->

<inherits name="edu.ucdenver.bios.webservice.common.common"/>

<!-- Specify the app entry point class. -->

<entry-point class='*entry point class*'/>

…

</module>

# Module and Component Descriptions

## Component overview

The domain layer allows GLIMMPSE modules to communicate using a common object language. The objects described below are all Plain Old Java Objects (Pojos). Each object provides getter/setter methods for each field, although these are omitted from this document for clarity. Any additional methods are outlined for each object below.

Objects are transmitted between the GLIMMPSE user interface and the web services layer via serialized Java objects.

### The StudyDesign Object

The StudyDesign object contains all study design information required for power and sample size analysis. This includes the type of calculation, predictor and response variables, and hypotheses. The StudyDesign object contains matrix representations of the study design, and meta information such as variable names, clustering hierarchy if applicable, etc.

The StudyDesign object contains the following fields and sub-objects.

|  |  |  |  |
| --- | --- | --- | --- |
| # | Variable | Type | Description |
|  | UUID | Byte[16] | A unique identifier 8 for the StudyDesign (see section 3.1.1.1) |
|  | Name | String | Name of the study design |
|  | participantLabel | String | The participant label for study design. |
|  | solutionTypeEnum | solutionTypeEnum | The solutionTypeEnum indicates whether the user is solving for power, sample size, or detectable difference. Valid values are  “Power”  ”Sample Size”  ”Detectable Difference”  This is an instance of SolutionTypeEnum object (see section 3.1.1.2) |
|  | viewTypeEnum | viewTypeEnum | The viewTypeEnum indicates whether the user is using guided mode, matrix mode or upload mode. Valid values are  “Guided Mode”  ”Matrix Mode”  ”Upload”  This is an instance of SolutionTypeEnum object (see section 3.1.1.3) |
|  | matrixSet | HashSet | Set containing all matrices required for a power or sample size calculation. The matrices are instances of NamedMatrix (see section 3.1.4) |
|  | betweenParticipantFactorList | Set | List of fixed predictor names and values. Each factor is an instance of the BetweenParticipantFactor object (see section 3.1.9 ) |
|  | gaussianCovariate | Boolean | If true, this flag indicates that the user wishes to control for a Gaussian covariate in their study design. |
|  | clusteringTree | Set | Describes the hierarchy of clustering for the study design via a set of ClusterNode objects. (see section 3.1.11) |
|  | repeatedMeasuresTree | Set | Describes nested repeated measures (i.e. singly, double, triply repeated, etc.) for the study design via a set of RepeatedMeasuresNode objects. (see section 3.1.13) |
|  | hypothesis | Set | Describes the primary study hypothesis via a Hypothesis object (see section 3.1.15). Only a single hypothesis is allowed for version 2.0.0. |
|  | covariance | Set | Covariance information for within subject factors and the Gaussian covariate. (see section 3.1.20). |
|  | confidenceIntervalDescriptionSet | Set | Inputs to produce a power confidence interval. The set contains instances of the ConfidenceIntervalDescription object (see section 3.1.25). For version 2.0.0, only a single confidence interval description is allowed. |
|  | powerCurveDescriptionSet | Set | Inputs for producing a power curve from the calculation results. The set contains instances of the PowerCurveDescription object (see section 3.1.23). For version 2.0.0, only a single power curve description is allowed. |
|  | alphaList | List | List of Type I error values. Each value is an instance of a TypeIError object (see section 3.1.27) |
|  | betaScaleList | List | List of beta scale values, which allow the user to calculate power and sample size for a variety of possible mean differences. Each value is an instance of a BetaScale object (see section 3.1.29). |
|  | sigmaScaleList | List | List of sigma scale values, which allow the user to calculate power and sample size for a variety of possible covariance values. Each value is an instance of a SigmaScale object (see section 3.1.31). |
|  | relativeGroupSizeList | List | List of relative group size values, assuming a cell means coding. Each value is an instance of a RelativeGroupSize object (see section 3.1.33). |
|  | statisticalTestList | List | List of statistical tests for which to calculate power and sample size. Each value is an instance of a StatisticalTestListobject (see section 3.1.35). |
|  | powerMethodList | List | List of power methods to use. Each value is an instance of the PowerMethod object (see section 3.1.37). |
|  | quantileList | List | List of quantiles associated with the quantile power method. Each value is an instance of the Quantile object (see section 3.1.39). |
|  | nominalPowerList | List | List of desired minimum power values. Only used when performing a sample size calculation. Each value is an instance of the NominalPower object (see section 3.1.41) |
|  | responseList | List | List of response variables names. Names are represented as strings. NOTE: These are the outcomes assessed at a single measurement episode. Outcomes representing repeated measures should be specified as part of the repeatedMeasuresTree object. Each value is an instance of a ResponseNode object (see section 3.1.43) |
|  | sampleSizeList | List | List of sample size values. Only used when performing a power calculation. Each value is an instance of a SampleSize object (see section 3.1.45) |

#### StudyDesign UUIDs

StudyDesign objects are uniquely identified by a UUID. Within the Study Design Service, the study design UUID is used to synchronize persistence of the StudyDesign object across multiple database tables. The GLIMMPSE user interface uses the UUID to update and retrieve data for a given StudyDesign object.

The StudyDesign UUID is a 16-byte (128-bit) long as described by Leach et al.8 Examples of UUIDs in hexadecimal:

067e6162-3b6f-4ae2-a171-2470b63dff00

54947df8-0e9e-4471-a2f9-9af509fb5889

The UUIDs are ‘practically unique’ rather than ‘guaranteed unique’. There are 1632= 340,282,366,920,938,463,463,374,607,431,768,211,456 possible UUIDs.

Thus the probability of creating a few tens of trillions of UUIDs in a year and having one duplicate is 0.00000000006.

#### The SolutionTypeEnum Object

Each study design object is associated with a Solution Type. This ENUM class enlists a variety of possible solution types. The SolutionTypeEnum object contains the following fields

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| POWER |  | It is an Enum instance of this object. It describes a solution type named ‘Power’. |
| SAMPLE\_SIZE |  | It is an Enum instance of this object. It describes a solution type named ‘Sample Size’. |
| DETECTABLE\_DIFFERENCE |  | It is an Enum instance of this object. It describes a solution type named ‘Detectable Difference’. |
| Id | String | Describes the id used to define a Enum instance of this object. For example, when id = ‘Power’ Enum instance = POWER. |

#### The StudyDesignViewTypeEnum Object

Each study design object is associated with a view type. This ENUM class enlists a variety of possible view types. The StudyDesignViewTypeEnum object contains the following fields

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| GUIDED\_MODE |  | It is an Enum instance of this object. It describes a solution type named ‘Guided Mode’. |
| MATRIX\_MODE |  | It is an Enum instance of this object. It describes a solution type named ‘Matrix Mode’. |
| UPLOAD |  | It is an Enum instance of this object. It describes a solution type named ‘Upload. |
| Id | String | Describes the id used to define a Enum instance of this object. For example, when id = ‘Upload’ Enum instance = UPLOAD. |

### The StudyDesignList Object

The StudyDesignList object describes List of StudyDesign objects.

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Uuid | Byte[16] | A unique identifier 8 for the StudyDesign (see section 3.1.1.1) |
| studyDesignList | List<StudyDesign> | This is a list of StudyDesign objects. (see section 3.1.1 ) |

This object is added as a work around for Jackson Serialization issues.

### The Blob2DArray Object

The Blob2DArray object describes a blob holder for some objects in the study design. Blob is a MySQL data type which holds arbitrary size data. The Blob2DArray object has the following fields;

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| data | Double[][] | 2D array of values of type double. |

### The NamedMatrix Object

The NamedMatrix object describes a named, *n*×*m* matrix. The NamedMatrix object has the following fields

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Id | Int | Primary identifier for the object (with respect to other NamedMatrix objects. Not universally unique) |
| name | String | Name of the matrix |
| rows | Int | Matrix row dimension |
| columns | Int | Matrix column dimension |
| data | Blob2DArray | Object which holds the matrix cell values as a double dimension array. The data object is instance of Blob2DArray object. (See section 3.1.2) |

### The NamedMatrixList object

The NamedMatrixList object describes a List of NamedMatrix objects. It extends ArrayList<NamedMatrix>.

This object is added as a work around for Jackson Serialization issues.

### The NamedMatrixSet object

The NamedMatrixSet object describes Set of NamedMatrix objects.

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Uuid | Byte[16] | A unique identifier 8 for the StudyDesign (see section 3.1.1.1) |
| matrixSet | Set<NamedMatrix> | Set containing all matrices required for a power or sample size calculation. The matrices are instances of NamedMatrix (see section 3.1.4 ) |

This object is added as a work around for Jackson Serialization issues.

### The UuidMatrix object

The UuidMatrix object describes following fields;

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Uuid | Byte[16] | A unique identifier 8 for the StudyDesign (see section 3.1.1.1) |
| matrix | NamedMatrix | A single matrix which is saved in/retrieved from a database. The matrix is instance of NamedMatrix (see section 3.1.4 ) |

This object is added as a work around for Jackson Serialization issues.

### The UuidMatrixName object

The UuidMatrixName object describes following fields;

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Uuid | Byte[16] | A unique identifier 8 for the StudyDesign (see section 3.1.1.1) |
| matrixName | String | The name of a matrix which is to be retrieved from database. |

This object is added as a work around for Jackson Serialization issues.

### The BetweenParticipantFactor Object

The BetweenParticipantFactor object describes a fixed predictor in the study design. Fixed predictors are determined by the study design. Values of fixed predictors are known prior to drawing a sample. The BetweenParticipantFactor object has the following fields

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Id | Int | Primary identifier for the object (with respect to other BetweenParticipantFactor objects. Not universally unique) |
| predictorName | String | Name of the predictor |
| categoryList | List<Category> | List of valid values for the predictor. (see section 3.1.9.1) |

#### The Category Object

The Category object describes categories for a predictor object (See section 3.1.9). The Category object has the following fields

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Id | Int | Primary identifier for the object (with respect to other Category objects. Not universally unique) |
| category | String | Name of the category. A BetweenParticipantFactor holds multiple categories (See section 3.1.9). |

#### The CategoryList Object

The CategoryList object describes List of Category objects.

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Id | int | A unique identifier for the Category (see section 3.1.9.2) |
| category | String | The category names. |

This object is added as a work around for Jackson Serialization issues.

### The BetweenParticipantFactorList Object

The BetweenParticipantFactorList object describes List of BetweenParticipantFactor objects.

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Uuid | Byte[16] | A unique identifier 8 for the StudyDesign (see section 3.1.1.1) |
| betweenParticipantFactorList | List<BetweenParticipantFactor> | List of fixed predictor names and values. Each factor is an instance of the BetweenParticipantFactor object (see section 3.1.9 ) |

This object is added as a work around for Jackson Serialization issues.

### The Cluster Node Object

In a multilevel study design, participants are organized into clusters. Observations on participants within a cluster are assumed to be correlated. Clustering may have one or more levels. Multilevel features of the study design are represented by a tree of ClusterNode objects. The ClusterNode describes clustering at a single level, and the tree determines the hierarchical organization of these nodes.

For example, consider a study design which examines the impact of a new reading program on standardized test scores. Suppose that 5 counties participate, sampling 10 schools within each county, and recruiting 100 students within each school. Suppose that the within county correlation is 0.01, and the within-school correlation is 0.005. The clustering for this design would be represented as follows

Level 1:

Name: County

Size: 10 schools

ICC: 0.01

Level 2:

Name: School

Size: 100 students

ICC: 0.005

For version 2.0.0, we assume equal cluster sizes at each level. The ClusterNode object has the following fields.

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Id | Int | Primary identifier for the object (with respect to other ClusterNode objects. Not universally unique) |
| groupName | String | Name of this clustering level. (ex. “school”, “census tract”, etc.) |
| groupSize | Int | Size of the cluster. For example, if a study includes 20 children within each school |
| intraClusterCorrelation | Double | Intra-cluster correlation. Value must be between -1 and 1. |
| node | Int | Position of the node in the clustering tree when traversed in depth-first order. For the root node, position = 1. |
| parent | Int | Position of the node’s parent in the clustering tree. |

### The ClusterNodeList Object

The ClusterNodeList object describes List of ClusterNode objects.

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Uuid | Byte[16] | A unique identifier 8 for the StudyDesign (see section 3.1.1.1) |
| clusterNodeList | List<ClusterNode> | Describes the hierarchy of clustering for the study design via a set of ClusterNode objects. (see section 3.1.11) |

This object is added as a work around for Jackson Serialization issues.

### The Repeated Measures Node Object

In a longitudinal study design, response variables are observed on multiple occasions or under multiple conditions for a each participant. Repeated measures may occur across multiple dimensions. For example, a study may measure weight each day for 1 month, and on each day measure weight in the morning and afternoon. Repeated measures features of the study design are represented by a tree of RepeatedMeasuresNode objects. The RepeatedMeasuresNode describes repeated measures information across a single dimension, and the tree determines the hierarchical organization of these nodes.

The RepeatedMeasuresNode object has the following fields.

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Id | Int | Primary identifier for the object (with respect to other RepeatedMeasuresNode objects. Not universally unique) |
| dimension | String | Name of the repeated measures dimension (ex. “week”) |
| repeatedMeasuresDimensionType | RepeatedMeasuresDimensionType | Indicates the type of dimension. Valid values are  “NUMERIC” – ratio or interval measurement scale. Allows unequal spacing.  “ORDINAL” – ordinal measurement scale  “NOMINAL” – nominal measurement scale (see section 3.1.13.2) |
| numberOfMeasurements | Int | Number of measurements for the current dimension |
| spacingList | List<Spacing> | For numeric values, a list of integers representing the spacing of measurements. For example, for measurements at 1,3, and 10 weeks, this list would contain (1,3,10). (see section 3.1.13.1) |
| position | Int | Position of the node in the clustering tree when traversed in depth-first order. For the root node, position = 1. |
| parent | Int | Position of the node’s parent in the clustering tree. |

Consider a study design which takes heart rate measurements on 20 subjects. Heart rate is measured at week 1, 3, and 10. Within each week, measurements are taken on Monday, Wednesday, and Friday. Lastly, within each day, heart rate is measured in supine, sitting, and standing. The following RepeatedMeasuresNode objects would be required to describe this study design

Dimension 1:

Name: week

Type: numeric

NumberOfMeasurements: 3

SpacingList: 1,3,10

Dimension 2:

Name: weekday

Type: ordinal

NumberOfMeasurements: 3

Dimension 3:

Name: body position

Type: nominal

NumberOfMeasurements: 3

#### The Spacing Object

The Spacing object describes spacing values for a RepeatedMeasuresNode object (See section 3.1.13). The Spacing object has the following fields

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Id | Int | Primary identifier for the object (with respect to other Category objects. Not universally unique) |
| Value | int | The spacing value for a RepeatedMeasuresNode object(See section 3.1.13). |

#### The RepeatedMeasuresDimensionType Object

Each study design object is associated with a RepeatedMeasuresDimensionType. This ENUM class enlists a variety of possible dimension types. The RepeatedMeasuresDimensionType object contains the following fields

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| NUMERICAL |  | It is an Enum instance of this object. It describes a solution type named ‘Numerical’. |
| ORDINAL |  | It is an Enum instance of this object. It describes a solution type named ‘Ordinal’. |
| CATEGORICAL |  | It is an Enum instance of this object. It describes a solution type named ‘Categorical’. |
| Id | String | Describes the id used to define a Enum instance of this object. For example, when id = ‘Categorical’ Enum instance = CATEGORICAL. |

### The RepeatedMeasuresNodeList Object

The RepeatedMeasuresNodeList object describes List of RepeatedMeasuresNode objects.

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Uuid | Byte[16] | A unique identifier 8 for the StudyDesign (see section 3.1.1.1) |
| repeatedMeasuresList | List< RepeatedMeasuresNode> | Describes the hierarchy of Repeated Measures for the study design via a set of RepeatedMeasuresNode objects. (see section 3.1.13) |

This object is added as a work around for Jackson Serialization issues.

### The Hypothesis Object

The Hypothesis Object describes the primary study hypothesis. Possible hypotheses include main effects, interactions, and trends. See the Full Factorial Hypothesis9 design document for details on supported hypotheses. The Hypothesis objects has the following fields

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Id | Int | Primary identifier for the object |
| type | HypothesisTypeEnum | Type of hypothesis. Valid values are  “Main effect” – tests the effect of a single covariate  “Interaction” – tests the interaction between two or more covariates  “Trend”  (see section 3.1.15.3) |
| betweenParticipantFactorList | List | List of between participant factors associated with the hypothesis. Each variable is an instance of a HypothesisBetweenParticipantFactor object (see section 3.1.15.1) |
| repeatedMeasuresNodeList | List | List of repeated measure dimensions involved in the hypothesis. Each variable is an instance of a HypothesisRepeatedMeasuresNode object (see section 3.1.15.2) |

#### The HypothesisBetweenParticipantFactor Object

The HypothesisBetweenParticipantFactor object is a map between a Hypothesis object and a BetweenParticipantFactor object included in the hypothesis of interest. The HypothesisBetweenParticipantFactor object has the following fields.

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| hypothesis | Hypothesis | Reference to the parent Hypothesis object. |
| betweenParticipantFactor | BetweenParticipantFactor | Reference to the BetweenParticipantFactor object |
| type | HypothesisTrendTypeEnum | Type of trend tested for this factor. Valid values are  “None”  “Change from baseline”  “All polynomial trends”  “Linear trend”  “Quadratic trend”  “Cubic trend”  (see section 3.1.15.4) |

#### The HypothesisRepeatedMeasuresNode Object

The HypothesisRepeatedMeasuresNode object is a map between a Hypothesis object and a RepeatedMeasuresNode object included in the hypothesis of interest. The HypothesisRepeatedMeasuresNode object has the following fields.

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| hypothesis | Hypothesis | Reference to the parent Hypothesis object. |
| repeatedMeasuresNode | RepeatedMeasuresNode | Reference to the RepeatedMeasuresNode object |
| type | HypothesisTrendTypeEnum | Type of trend tested for this factor. Valid values are  “None”  “Change from baseline”  “All polynomial trends”  “Linear trend”  “Quadratic trend”  “Cubic trend”  (see section 3.1.15.4) |

#### The HypothesisTypeEnum Object

Each study design object is associated with a HypothesisTypeEnum. This ENUM class enlists a variety of possible hypothesis types. The HypothesisTypeEnum object contains the following fields

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| MAIN\_EFFECT |  | It is an Enum instance of this object. It describes a solution type named ‘Main Effect’. |
| INTERACTION |  | It is an Enum instance of this object. It describes a solution type named ‘Interaction’. |
| TREND |  | It is an Enum instance of this object. It describes a solution type named ‘Trend’. |
| Id | String | Describes the id used to define a Enum instance of this object. For example, when id = ‘Trend’ Enum instance = TREND. |

#### The HypothesisTrendTypeEnum Object

Each study design object is associated with a HypothesisTrendTypeEnum. This ENUM class enlists a variety of possible hypothesis trend types. The HypothesisTrendTypeEnum object contains the following fields

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| NONE |  | It is an Enum instance of this object. It describes a solution type named ‘None’. |
| CHANGE\_FROM\_BASELINE |  | It is an Enum instance of this object. It describes a solution type named ‘Change from baseline’. |
| ALL\_POYNOMIAL |  | It is an Enum instance of this object. It describes a solution type named ‘All polynomial trends’. |
| LINEAR |  | It is an Enum instance of this object. It describes a solution type named ‘Linear’. |
| QUADRATIC |  | It is an Enum instance of this object. It describes a solution type named ‘Quadratic. |
| CUBIC |  | It is an Enum instance of this object. It describes a solution type named ‘Cubic’. |
| Id | String | Describes the id used to define a Enum instance of this object. For example, when id = ‘Cubic’ Enum instance = CUBIC. |

### The HypothesisSet Object

The HypothesisSet object describes Set of Hypothesis objects.

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Uuid | Byte[16] | A unique identifier 8 for the StudyDesign (see section 3.1.1.1) |
| hypothesisSet | Set<Hypothesis> | Set containing all hypothesis required for a power or sample size calculation. The hypothesis are instances of Hypothesis object (see section 3.1.15 ) |

This object is added as a work around for Jackson Serialization issues.

### The UuidHypothesis Object

The UuidHypothesis object describes following fields;

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Uuid | Byte[16] | A unique identifier 8 for the StudyDesign (see section 3.1.1.1) |
| hypothesis | Hypothesis | A single hypothesis which is saved in/retrieved from a database. The hypothesis is instance of Hypothesis object (see section 3.1.15 ) |

This object is added as a work around for Jackson Serialization issues.

### The UuidHypothesisType Object

The UuidHypothesis object describes following fields;

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Uuid | Byte[16] | A unique identifier 8 for the StudyDesign (see section 3.1.1.1) |
| type | HypothesisTypeEnum | Type of hypothesis. Valid values are  “Main effect” – tests the effect of a single covariate  “Interaction” – tests the interaction between two or more covariates  “Trend”  (see section 3.1.15.3) |

This object is added as a work around for Jackson Serialization issues.

### The Covariance Object

The GLIMMPSE system can account for variability from the following sources.

* Between participant correlation due to clustering, i.e. intra-cluster correlation
* Within participant covariance due to repeated measures
* Covariance between outcomes and a Gaussian covariate

Note that intra-cluster correlation is described by the ClusterNode object (see section 3.1.11).

Covariance information can be represented in three difference forms.

1. Lear Model10. A structured covariance model with three parameters describing the standard deviation, the correlation for measurements a minimum distance apart, and a rate of decay of the correlation as distance between measurements increases
2. Unstructured correlation. Requires specification of the standard deviation of each variable and the upper triangle (excluding the diagonal) of the correlation matrix for the variables.
3. Unstructured covariance. Requires specification of the upper triangle (including the diagonal) of the covariance matrix.

To capture the information in each of these forms, the Covariance object will have the following fields

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Id | Int | Primary identifier for the object |
| Name | String | Name of the covariance matrix |
| standardDeviationList | List<standardDeviation> | Standard deviation for Lear model or unstructured correlation (see section 3.1.19.1) |
| rho | Double | The rho parameter of the Lear model |
| delta | Double | Delta parameter of the Lear model |
| Rows | Int | Number of rows in the covariance matrix |
| Columns | Int | Number of columns in the covariance matrix |
| data | Blob2DArray | Object which holds the matrix cell values as a double dimension array. The data object is instance of Blob2DArray object. (See section 3.1.3) |

#### The StandardDeviation Object

The StandardDeviation object describes standard deviation for a covariance object (See section 3.1.19). The Category object has the following fields

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Id | Int | Primary identifier for the object (with respect to other StandardDeviation objects. Not universally unique) |
| value | double | Value of the standard deviation. A Covariance object holds multiple categories (See section 3.1.19). |

### The CovarianceSet Object

The CovarianceSet object describes Set of Covariance objects.

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Uuid | Byte[16] | A unique identifier 8 for the StudyDesign (see section 3.1.1.1) |
| CovarianceSet | Set< Covariance> | Set containing all covariance required for a power or sample size calculation. The covariance are instances of Covariance object (see section 3.1.19 ) |

This object is added as a work around for Jackson Serialization issues.

### The UuidCovariance Object

The UuidCovariance object describes following fields;

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Uuid | Byte[16] | A unique identifier 8 for the StudyDesign (see section 3.1.1.1) |
| covariance | Covariance | A single covariance which is saved in/retrieved from a database. The covariance is instance of Covariance object (see section 3.1.19 ) |

This object is added as a work around for Jackson Serialization issues.

### The UuidCovarianceName Object

The UuidCovarianceName object describes following fields;

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Uuid | Byte[16] | A unique identifier 8 for the StudyDesign (see section 3.1.1.1) |
| covarianceName | String | The name of a covariance which is to be retrieved from database. |

This object is added as a work around for Jackson Serialization issues.

### The PowerCurveDescription Object

Upon completion of a power or sample size calculation, the user may request the creation of a power curve. The PowerCurveDescription object contains information required for drawing power curve. The object the following fields:

|  |  |  |  |
| --- | --- | --- | --- |
| # | Field Name | Field Type | Description |
|  | Id | Int | Primary identifier for the object |
|  | title | String | Title for the plot |
|  | horizontalAxisLabelEnum | HorizontalAxisLabelEnum | Indicates the value to plot on the horizontal axis. Valid values are  “Total Sample Size”  “Regression Coefficient Scale Factor”  “Variability Scale Factor”  (see section 3.1.23.1) |
|  | stratificationVarEnum | StratificationVarEnum | Indicates the stratification variable. Separate curves will be displayed for each value of the specified stratification variable. Valid values are  “Statistical Test”  “Total Sample Size”  “Regression Coefficient Scale Factor”  “Variability Scale Factor”  “Type I Error”  “Power Method”  “Quantile” (only available for quantile power method)  (see section 3.1.23.2 ) |
|  | statisticalTestTypeEnum | StatisticalTestTypeEnum | Only power values for the specified test will be included in the plot. Ignored if “Statistical Test” is the stratification variable.  (see section 3.1.23.3) |
|  | typeIError | Double | Only power values for the specified Type I Error level will be included in the plot. Ignored if “Type I Error” is the stratification variable. |
|  | sampleSize | Int | Only power values for the specified sample size will be included in the plot. Ignored if “Total Sample Size” is the stratification variable or horizontal axis type. |
|  | betaScale | Double | Only power values for the specified beta scale level will be included in the plot. Ignored if “Regression Coefficient Scale Factor” is the stratification variable or horizontal axis type. |
|  | sigmaScale | Double | Only power values for the specified sigma scale level will be included in the plot. Ignored if “Variability Scale Factor” is the stratification variable or horizontal axis type. |
|  | powerMethodEnum | PowerMethodEnum | Only power values for the specified power method will be included in the plot. Ignored if “Power Method” is the stratification variable.  (see section 3.1.23.4) |
|  | quantile | Double | Only power values for the specified quantile will be included in the plot. Ignored if “Quantile” is the stratification variable. |

#### The HorizontalAxisLabelEnum Object

Each study design object is associated with a HorizontalAxisLabelEnum. This ENUM class enlists a variety of possible Horizontal Axis Label types. The HorizontalAxisLabelEnum object contains the following fields;

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| VARIABILITY\_SCALE\_FACTOR |  | It is an Enum instance of this object. It describes a solution type named ‘Variability Scale Factor’. |
| TOTAL\_SAMPLE\_SIZE |  | It is an Enum instance of this object. It describes a solution type named ‘Total Sample Size’. |
| REGRESSION\_COEEFICIENT\_SCALE\_FACTOR |  | It is an Enum instance of this object. It describes a solution type named ‘Regression Coefficient Scale Factor’. |
| Id | String | Describes the id used to define a Enum instance of this object. For example, when id = ‘Regression Coefficient Scale Factor’ Enum instance = REGRESSION\_COEEFICIENT\_SCALE\_FACTOR. |

#### The StratificationVariableEnum Object

Each study design object is associated with a StratificationVariableEnum. This ENUM class enlists a variety of possible Stratification Variable types. The StratificationVariableEnum object contains the following fields;

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| TYPE\_I\_ERROR |  | It is an Enum instance of this object. It describes a solution type named ‘Type I error’. |
| STATISTICAL\_TEST |  | It is an Enum instance of this object. It describes a solution type named ‘Statistical Test’. |
| Id | String | Describes the id used to define a Enum instance of this object. For example, when id = ‘Statistical Test’ Enum instance = STATISTICAL\_TEST. |

#### The PowerMethodEnum Object

Each study design object is associated with a PowerMethodEnum. This ENUM class enlists a variety of possible Power Method types. The PowerMethodEnum object contains the following fields;

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| CONDITIONAL |  | It is an Enum instance of this object. It describes a solution type named ‘Conditional’. |
| UNCONDITIONAL |  | It is an Enum instance of this object. It describes a solution type named ‘Unconditional’. |
| QUANTILE |  | It is an Enum instance of this object. It describes a solution type named ‘Quantile’. |
| Id | String | Describes the id used to define a Enum instance of this object. For example, when id = ‘Conditional’ Enum instance = CONDITIONAL. |

#### The StatisticalTestTypeEnum Object

Each study design object is associated with a StatisticalTestTypeEnum. This ENUM class enlists a variety of possible Statistical Test types. The StatisticalTestTypeEnum object contains the following fields

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| UNIREP |  | It is an Enum instance of this object. It describes a solution type named ‘unirep’. |
| UNIREPBOX |  | It is an Enum instance of this object. It describes a solution type named ‘unirepBox’. |
| UNIREPGG |  | It is an Enum instance of this object. It describes a solution type named ‘unirepGG’. |
| UNIREPHF |  | It is an Enum instance of this object. It describes a solution type named ‘unirepHF’. |
| WL |  | It is an Enum instance of this object. It describes a solution type named ‘wl’. |
| PBT |  | It is an Enum instance of this object. It describes a solution type named ‘pbt’. |
| HLT |  | It is an Enum instance of this object. It describes a solution type named ‘hlt’. |
| Id | String | Describes the id used to define a Enum instance of this object. For example, when id = ‘Statistical Test’ Enum instance = UNIREP. |

### The UuidPowerCurveDescription Object

The UuidPowerCurveDescription object describes following fields;

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Uuid | Byte[16] | A unique identifier 8 for the StudyDesign (see section 3.1.1.1) |
| powerCurveDescription | PowerCurveDescription | A single powerCurveDescription which is saved in/retrieved from a database. The covariance is instance of PowerCurveDescription object (see section 3.1.23 ) |

This object is added as a work around for Jackson Serialization issues.

### The ConfidenceIntervalDescription Object

The ConfidenceIntervalDescription object includes information required to produce a confidence interval on power values. The confidence intervals are produced using the methods described by Taylor and Muller11. When determining values for mean differences and covariance in a power and sample size analysis, scientists typically use data from prior research. We refer to this data as the pilot data. Since the estimated means and standard deviations from the pilot data have a degree of uncertainty, the power values will also contain some uncertainty. The ConfidenceIntervalDescription object includes fields which describe the uncertainty from the pilot data, and the width of the desired confidence interval for power values.

|  |  |  |
| --- | --- | --- |
| Variable | Type | Description |
| Id | Int | Primary identifier for the object |
| betaFixed | Boolean | If true, the estimated mean values from the pilot data set are assumed certain and fixed. |
| sigmaFixed | Boolean | If true, the estimated covariance values from the pilot data set are assumed certain and fixed. |
| lowerTailProbability | Double | Lower tail probability for the confidence interval |
| upperTailProbability | Double | Upper tail probability for the confidence interval |
| sampleSize | Int | Sample size of the pilot data from which the beta and sigma values were obtained |
| rankOfDesignMatrix | Int | Rank of the design matrix for the model used to analyze the pilot data |

### The UuidConfidenceIntervalDescription Object

The UuidConfidenceIntervalDescription object describes following fields;

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Uuid | Byte[16] | A unique identifier 8 for the StudyDesign (see section 3.1.1.1) |
| confidenceInterval | ConfidenceInterval | A single confidenceInterval which is saved in/retrieved from a database. The confidenceInterval is instance of ConfidenceIntervaDescription object (see section 3.1.25 ) |

This object is added as a work around for Jackson Serialization issues.

### The TypeIError Object

The TypeIError object is a wrapper class for Type I Error (α) values. It contains the following fields

|  |  |  |
| --- | --- | --- |
| Variable | Type | Description |
| Id | Int | Primary identifier for the object |
| alphaValue | Double | Type I Error value. Must be between 0 and 1. |

### The TypeIErrorList Object

The TypeIErrorList object describes List of TypeIError objects.

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Uuid | Byte[16] | A unique identifier 8 for the StudyDesign (see section 3.1.1.1) |
| typeIErrorList | List<TypeIError> | This is a list of TypeIError objects. (see section 3.1.27 ) |

This object is added as a work around for Jackson Serialization issues.

### The BetaScale Object

The BetaScale object is a wrapper class for beta scale values. It contains the following fields

|  |  |  |
| --- | --- | --- |
| Variable | Type | Description |
| Id | Int | Primary identifier for the object |
| value | Double | Beta scale value. Must be a positive value. |

### The BetaScaleList Object

The BetaScaleList object describes List of BetaScale objects.

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Uuid | Byte[16] | A unique identifier 8 for the StudyDesign (see section 3.1.1.1) |
| BetaScaleList | List< BetaScale > | This is a list of BetaScale objects. (see section 3.1.29 ) |

This object is added as a work around for Jackson Serialization issues.

### The SigmaScale Object

The SigmaScale object is a wrapper class for beta scale values. It contains the following fields

|  |  |  |
| --- | --- | --- |
| Variable | Type | Description |
| Id | Int | Primary identifier for the object |
| value | Double | Sigma scale value. Must be a positive value. |

### The SigmaScaleList Object

The SigmaScaleList object describes List of SigmaScale objects.

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Uuid | Byte[16] | A unique identifier 8 for the StudyDesign (see section 3.1.1.1) |
| SigmaScaleList | List< SigmaScale> | This is a list of SigmaScaleobjects. (see section 3.1.31 ) |

This object is added as a work around for Jackson Serialization issues.

### The RelativeGroupSize Object

The RelativeGroupSize object is a wrapper class for relative group size values. The value describes the size of the current group relative to the smallest group in the study. It contains the following fields

|  |  |  |
| --- | --- | --- |
| Variable | Type | Description |
| Id | Int | Primary identifier for the object |
| value | Int | Relative group size value. Must be greater than 0. |

### The RelativeGroupSizeList Object

The RelativeGroupSizeList object describes List of RelativeGroupSize objects.

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Uuid | Byte[16] | A unique identifier 8 for the StudyDesign (see section 3.1.1.1) |
| RelativeGroupSizeList | List< RelativeGroupSize> | This is a list of RelativeGroupSize objects. (see section 3.1.33 ) |

This object is added as a work around for Jackson Serialization issues.

### The StatisticalTest Object

The Test object describes the statistical test for which power or sample size is calculated. Several approximation methods and tests are available. Test type must be specified, but all approximation methods are optional. The Test object has the following fields

|  |  |  |
| --- | --- | --- |
| Variable | Type | Description |
| Id | Int | Primary identifier for the object |
| type | StatisticalTestTypeEnum | Statistical test. Valid values are:  “unirep” – Univariate approach to repeated measures, assuming sphericity (uncorrected)  “unirepBox” - Univariate approach to repeated measures with Box correction  “unirepGG” - Univariate approach to repeated measures with Geisser-Greenhouse correction  “unirepHF” - Univariate approach to repeated measures with Huynh-Feldt correction  “wl” – Wilk’s Lambda test (multivariate)  “pbt” – Pillai Bartlett Trace test (multivariate)  “hlt” – Hotelling Lawley Trace (multivariate)  (See section 3.1.23.4) |

### The StatisticalTestList Object

The StatisticalTestList object describes List of StatisticalTest objects.

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Uuid | Byte[16] | A unique identifier 8 for the StudyDesign (see section 3.1.1.1) |
| StatisticalTestList | List< StatisticalTest> | This is a list of StatisticalTest objects. (see section 3.1.35 ) |

This object is added as a work around for Jackson Serialization issues.

### The PowerMethod Object

The PowerMethod object is a wrapper class for power calculation methods. It contains the following fields

|  |  |  |
| --- | --- | --- |
| Variable | Type | Description |
| Id | Int | Primary identifier for the object |
| powerMethodEnum | PowerMethodEnum | Power calculation method. Valid values are  “conditional”  “unconditional”  “quantile”  (see section 3.1.23.3) |

### The PowerMethodList Object

The RelativeGroupSizeList object describes List of RelativeGroupSize objects.

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Uuid | Byte[16] | A unique identifier 8 for the StudyDesign (see section 3.1.1.1) |
| RelativeGroupSizeList | List< RelativeGroupSize> | This is a list of RelativeGroupSize objects. (see section 3.1.37 ) |

This object is added as a work around for Jackson Serialization issues.

### The Quantile Object

The Quantile object is a wrapper class for quantile values associated with the quantile power method. It contains the following fields

|  |  |  |
| --- | --- | --- |
| Variable | Type | Description |
| Id | Int | Primary identifier for the object |
| value | Double | Quantile value. Must be between 0 and 1. |

### The QuantileList Object

The QuantileList object describes List of Quantile objects.

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Uuid | Byte[16] | A unique identifier 8 for the StudyDesign (see section 3.1.1.1) |
| QuantileList | List< Quantile> | This is a list of Quantile objects. (see section 3.1.39 ) |

This object is added as a work around for Jackson Serialization issues.

### The NominalPower Object

The NominalPower object is a wrapper class for nominal power values associated with a sample size calculation. It contains the following fields

|  |  |  |
| --- | --- | --- |
| Variable | Type | Description |
| Id | Int | Primary identifier for the object |
| value | Double | Nominal power value. Must be between 0 and 1. |

### The NominalPowerList Object

The NominalPowerList object describes List of NominalPower objects.

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Uuid | Byte[16] | A unique identifier 8 for the StudyDesign (see section 3.1.1.1) |
| NominalPowerList | List< NominalPower> | This is a list of NominalPower objects. (see section 3.1.41 ) |

This object is added as a work around for Jackson Serialization issues.

### The ResponseNode Object

The ResponseNode object is a wrapper class for response values associated with a sample size calculation. It contains the following fields

|  |  |  |
| --- | --- | --- |
| Variable | Type | Description |
| Id | Int | Primary identifier for the object |
| name | String | Response value. |

### The ResponseList Object

The ResponseList object describes List of ResponseNode objects.

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Uuid | Byte[16] | A unique identifier 8 for the StudyDesign (see section 3.1.1.1) |
| ResponseList | List< ResponseNode> | This is a list of ResponseNode objects. (see section 3.1.44 ) |

This object is added as a work around for Jackson Serialization issues.

### The SampleSize Object

The SampleSize object is a wrapper class for sample size values. It contains the following fields

|  |  |  |
| --- | --- | --- |
| Variable | Type | Description |
| Id | Int | Primary identifier for the object |
| value | Int | Must be minimum 1. |

### The SampleSizeList Object

The SampleSizeList object describes List of SampleSize objects.

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Uuid | Byte[16] | A unique identifier 8 for the StudyDesign (see section 3.1.1.1) |
| SampleSizeList | List< SampleSizeList> | This is a list of SampleSizeList objects. (see section 3.1.46 ) |

This object is added as a work around for Jackson Serialization issues.

### The ConfidenceInterval Object

The ConfidenceInterval object includes information required to produce a confidence interval on power values. The confidence intervals are produced using the methods described by Taylor and Muller11. When determining values for mean differences and covariance in a power and sample size analysis, scientists typically use data from prior research. We refer to this data as the pilot data. Since the estimated means and standard deviations from the pilot data have a degree of uncertainty, the power values will also contain some uncertainty. The ConfidenceInterval object includes following fields;

|  |  |  |
| --- | --- | --- |
| Variable | Type | Description |
| lowerLimit | Double | lower limit of the confidence interval. |
| upperLimit | Double | upper limit of the confidence interval. |
| alphaLower | Double | Lower tail probability for the confidence interval |
| alphaUpper | Double | Upper tail probability for the confidence interval |

### The FixedRandomMatrix Object

This object provides a matrix which contains fixed and random components. The combined matrix may be produced by concatenating the fixed and random submatrices either vertically or horizontally. The FixedRandomMatrix object has the following fields;

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| name | String | Name of the matrix |
| fixedMatrix | NamedMatrix | It is the fixed submatrix part of this object. (see section 3.1.4) |
| randomMatrix | NamedMatrix | It is the random submatrix part of this object. (see section 3.1.4) |
| combineHorizontal | boolean | Flag representing combine horizontal. |

### The PowerResult Object

This object contains a description of the general linear model power result

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| nominalPower | NominalPower | If solving for sample size, this is the target power  desired. Otherwise, equals the actual power  (see section 3.1.41) |
| actualPower | double | Used while calculated power |
| totalSampleSize | int | Total sample size for all groups |
| alpha | TypeIError | Type I error rate (see section 3.1.27) |
| betaScale | BetaScale | Scale factor for beta matrix  (see section 3.1.29) |
| sigmaScale | SigmaScale | Scale factor for the sigma error matrix  (see section 3.1.31) |
| test | StatisticalTest | Statistical test performed (see section 3.1.35) |
| powerMethod | PowerMethod | Power method used  (see section 3.1.37) |
| Quantile | Quantile | Quantile if using quantile power, null otherwise (see section 3.1.39) |
| confidenceInterval | ConfidenceInterval | Confidence limits for power if requested  only available if solving for power in a random design  (see section 3.1.47) |

### The PowerResultList Object

The PowerResultList object describes a List of PowerResult objects. It extends ArrayList<PowerResult>.

This object is added as a work around for Jackson Serialization issues.

### The StudyNamedMatrixList Object

The StudyNamedMatrixList object describes List of NamedMatrix objects.

|  |  |  |
| --- | --- | --- |
| Field Name | Field Type | Description |
| Uuid | Byte[16] | A unique identifier 8 for the StudyDesign (see section 3.1.1.1) |
| matrixList | List<NamedMatrix> | This is a list of NamedMatrix objects. (see section 3.1.39 ) |

This object is added as a work around for Jackson Serialization issues.

### The GLMMPower Object

The GLMMPower object contains the results of a single power calculation. Most requests involve multiple power calculations so GLMMPower objects are typically packaged into a list.

The GLMMPower object contains the following fields

|  |  |  |
| --- | --- | --- |
| Variable | Type | Description |
| alpha | Real | Type I error rate associated with this power result. The value ranges from 0 to 1, although 0.01, 0.05, and 0.10 are typical. |
| nominalPower | Real | The desired power for this calculation (specified with sample size or detectable difference requests). For power requests, this value will be equal to the actualPower attribute. |
| actualPower | Real | The actual power associated with this power result. For sample size and detectable difference requests, it may not be possible to perfectly match the desired power for certain study designs. This value will be set to the calculated power associated with the sample size or detectable difference which most closely matches the desired power. |
| sampleSize | Real | The total sample size for this power result |
| betaScale | Real | The scale factor applied to the beta matrix for this power result. This value can be used to generate the beta matrix representing the detectable difference. |
| sigmaScale | Real | The scale factor applied to the error matrix for this power result |
| powerMethod | Enum | Method used to calculate power. Valid values are condition, unconditional and quantile |
| Quantile | Real | When the power method is quantile, this is the quantile associated with the current power value |
| powerLower | Real | Confidence interval lower limit for the power value |
| powerUpper | Real | Confidence interval upper limit for the power value |

# [[1]](#footnote-1)

1. [↑](#footnote-ref-1)