

nordicAktiva v2.1

Paradigm Syntax Description

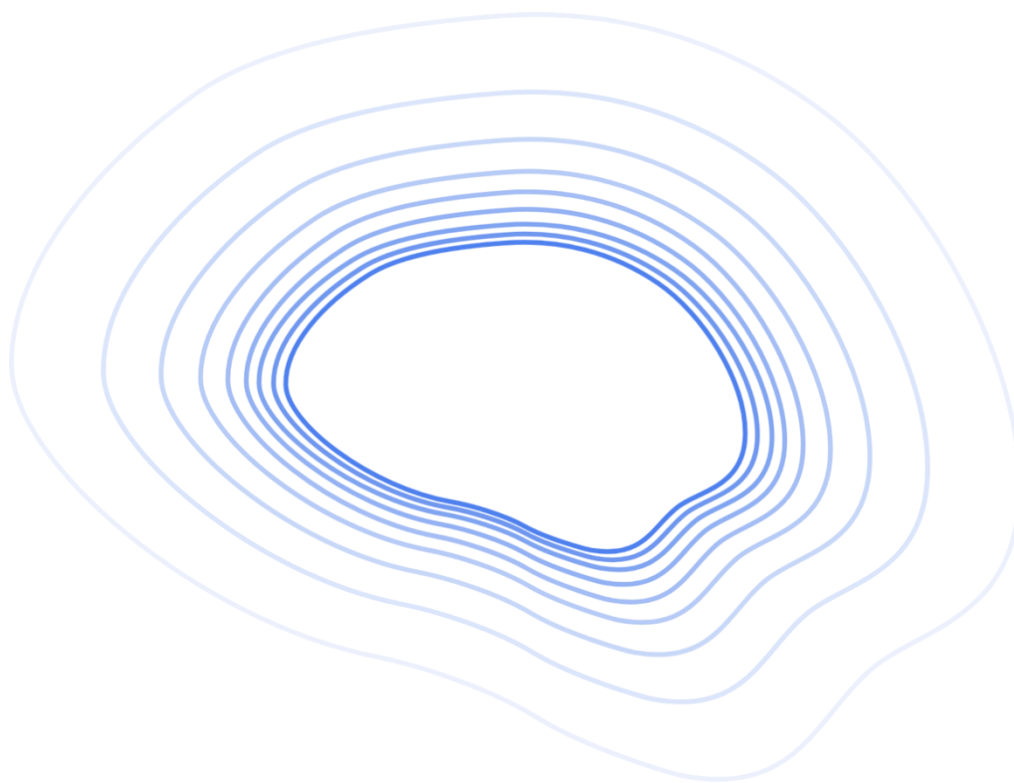


Table of Contents

1	Introduction.....	3
2	Viewing and editing the paradigm code.....	3
3	Structure.....	4
4	Main Settings.....	5
4.1	Variables.....	5
4.2	Settings.....	5
4.3	Languages.....	6
4.4	Description.....	6
4.5	Versions.....	7
4.6	Defaults.....	7
5	Definitions.....	8
5.1	Colors.....	8
5.2	Positions.....	9
5.3	Conditions.....	9
6	BOLDSettings.....	10
7	Stimuli.....	11
7.1	Slides.....	11
7.1.1	Expected Response and 'Clear' command.....	12
7.2	Trials.....	13
7.2.1	Start trial.....	14
7.2.2	Register key.....	14
7.2.3	Abort.....	14
7.2.4	Overlap.....	15
7.3	Blocks.....	15
7.4	Session.....	16
8	Resources.....	17

1 Introduction

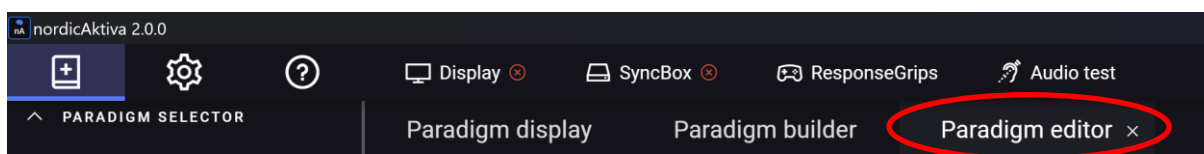
This document accompanies nordicAktiva v2.1 stimulus presentation software. It provides a description of the structure of paradigms and the syntax of the paradigm code.

Where possible, modifications to paradigms should be done using the Paradigm Builder user interface, therefore in most cases it is not necessary to work directly with the paradigm code, but this guide has been made available to help when needed.

Please note, it is crucial that any changes are checked thoroughly by previewing the paradigm in full prior to use on a subject.

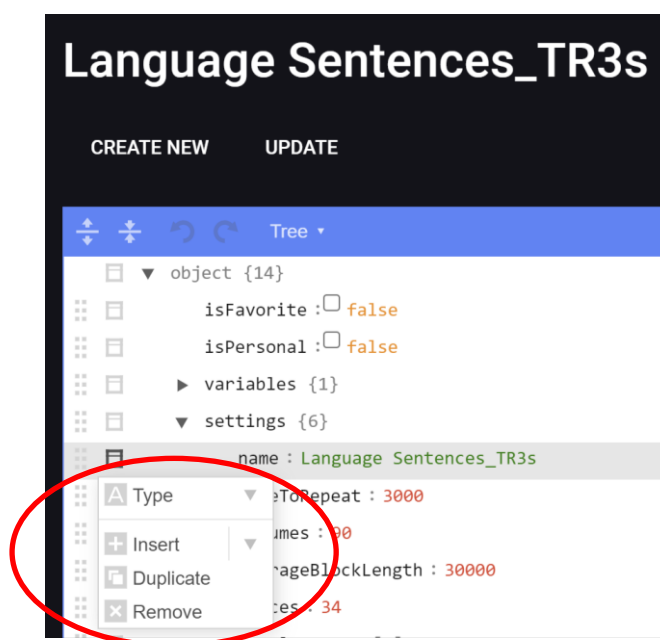
2 Viewing and editing the paradigm code

The code which dictates the content and timings of the paradigm is available to view in JSON format. A built-in JSON Editor allows the user to view and edit the JSON file by clicking the button 'Open in JSON Editor' from the nordicAktiva Paradigm Builder. This opens a 'Paradigm editor' tab in addition to the 'Paradigm Builder' and 'Paradigm Display' tabs, and these can be used in parallel by clicking on the required tab at any time.



A single file contains all the information necessary for the paradigm. A full description of all parts is included in the sections below. The order of the different parts of code is not important, and can be changed.

By default, the JSON editor shows the code in a 'Tree' view, which provides easy navigation through the paradigm and its attributes. The content can be changed by typing directly in the JSON Editor, or using the menu buttons in the columns to the left where appropriate:



If a change is made, the 'Update' button can be used to save the change, or the 'Create New' button can be used to create a copy with the change.

NOTE: Users will be prevented from making changes to NNL paradigms. Ideally a copy should be made by first duplicating an NNL paradigm in the main library, then opening it for editing. See User Manual for full details.

3 Structure

Paradigms contain several attributes which need to be defined. Some are required for the paradigm to run, others are optional to add extended functionality or flexibility.

The paradigm can be roughly grouped into 4 main parts that again might have several sub-parts consisting of one or more defined tags.

1. Main Settings
 - a. Variables
 - b. Settings (required)
 - c. Languages
 - d. Descriptions
 - e. Versions
 - f. Defaults
2. Definitions
 - a. Colors
 - b. Positions
 - c. Conditions
3. Stimulus
 - a. Slides (required)
 - b. Trials (required)
 - c. Blocks
 - d. Session (required)

4. Post-processing Definitions

The following sections take each of these 4 parts in detail and describe the syntax and dependencies.

4 Main Settings

4.1 Variables

Variables can be defined with a name and a value, and these variables can then be used at any place within the paradigm. This can be useful if a specific value (such as duration of a trial, or number of repetitions) is used several times within a paradigm. To alter the value everywhere in the paradigm, it would then be sufficient to change the value of the defined variable, not every instance.

When using the variable, a \$ sign should be placed in front of the variable name in the JSON file.

For example:

Definition of variable 'OnShowDuration':

```
▼ variables {1}
  OnShowDuration : 5000
```

Use of variable to define the duration of a slide to be shown within a trial:

```
▼ trial [49]
  ► 0 {4}
  ► 1 {2}
  ► 2 {2}
  ▼ 3 {2}
    name : S3
    ▼ show [1]
      ▼ 0 {2}
        duration : $OnShowDuration
        item : S3
```

4.2 Settings

In the Settings, the following general information about a paradigm is listed:

- **Name** Name of the paradigm to appear in nordicAktiva

Note: The name of each paradigm must be unique. The name of the JSON file in the local file system must match the paradigm name as defined in the Settings. See User Manual section on Importing Paradigms for further details.

- **timeToRepeat:** The Repetition Time (TR) for the associated EPI sequence on the MR scanner
- **Volumes:** Number of volumes (or dynamics / measurements) for the

associated EPI sequence on the MR scanner

- **AverageBlockLength** Average length of blocks in the paradigm. This is relevant when running with GE BrainWave, where the sync pulse markers only arrive when changing between on/off blocks.

- **Slices** The number of slices to be acquired in the image volume.

Note: For the majority of cases, this is not a relevant parameter, hence there is no user interface component for entering the number of slices. In some very rare cases, a scanner may send a sync pulse marker with every RF pulse, therefore with every slice. In this case, the number of slices should be set appropriately using the JSON editor, to communicate to the SyncBox how many sync pulse markers to count per volume of the EPI sequence. Furthermore, it can be useful if a design file is required which includes Slice Timing Correction.

- **Sample images** For NNL paradigms, images are provided as a guide to demonstrate the patterns of activation a user may expect to see in their resulting T-Maps. This section defines the file path to the sample images, and a caption.

Note: When a copy of a paradigm is created, it becomes a User paradigm, and the sample images are no longer included.

4.3 Languages

Different languages can be specified within nordicAktiva. Under 'Languages', all the implemented languages of the paradigm are listed.

nordicAktiva checks if more than one language is available and displays the choices via a drop down menu in the main Paradigm Display screen once a paradigm is selected. The default language which appears when any paradigm is selected is governed by the Global Settings (see User Manual for more information).

The selection of a given language calls the Description and Instructions (see below) for that language for the operator, and shows any language-specific (eg text) slides in the selected language during the paradigm run.

4.4 Description

For each language listed under 'Languages', the text associated with the paradigm Description and the paradigm Instructions in that language are written under 'Descriptions'.

Description should be a short explanation of the design of the paradigm, e.g. the number and duration and order of active and rest blocks, and also the main purpose of the paradigm.

Instructions is a prompt to the operator regarding how they should instruct the subject to participate in the paradigm. This includes a brief description of what the subject will see / hear, and how they should respond, if necessary.

The syntax is as follows:

```
Paradigm name\n\nDescription:\n Description \n---\nInstructions:\n Instructions \n---\n
```

4.5 Versions

The NNL paradigms are provided in four different versions which relate to our specific display hardware, due to the need for different sets of slides appropriate for different display resolutions. Full details of the resolution of the NNL displays are given in the nordicAktiva User Manual.

Versions listed within these parts of the code will be available to select from the main user interface 'Select version' drop down menu. They are also available to select from the Paradigm Builder when creating slides.

It is possible to add bespoke versions in the JSON editor. In the Tree view, the Actions menu can be used to duplicate one of the provided versions, and the name can be changed by typing new next into the field. Alternatively, existing versions can simply be renamed.

Please note that the pre-defined versions contain information about the resolution and aspect ratio which is appropriate to that version. As such, the preview of slides in the Builder, as well as the paradigm preview, will adjust its own aspect ratio to what the software expects. For user defined versions, it is best to duplicate or rename a provided version which has a similar resolution and aspect ratio to the patient display being used. When the slides are created and a preview is run, the user can then best visualize the correct layout of content on the slides.

4.6 Defaults

The Slide Builder user interface provides a 'Default' slide (see User Manual) where the user can specify the general appearance of e.g. fonts, colours, and the size of pictures. Within the JSON editor these sit under 'Defaults'.

Paradigms imported from nordicAktiva v1.3 will have their own default language and version, but within nordicAktiva v2.1 the Global Settings defaults are used for these parameters.

Default parameters include:

backgroundColor (color of the slide)

foregroundColor (color of the text)

fontFamily (style of the font)

fontSize (size of the font)

defaultPosition (in other slides, if no specific position is specified, what will be the position of the content)

defaultPicture (if a picture is imported, should it have a default size and/or position)

- width
- height
- horizontal (position)
- vertical (position)

Note that as per the example below, the font size and default picture are specific to the version:

```

▼ defaults {7}
  backgroundColor : Black
  defaultPosition : Default position
  foregroundColor : rgb(0,128,255)
  font : Arial
▼ versions {4}
  ► LCD-4K {1}
  ▼ LCD-HD {2}
    fontSize : 72
    ▼ defaultPicture {6}
      value : ./pictures/LeftHand.png
      horizontal : 267.272727272725
      vertical : 127.05794591504285
      height : 537.06688773307
      width : 560.4176219823329

```

5 Definitions

5.1 Colors

Colors are defined at the slide level, or as default colors under Defaults. The Slide Builder interface is the best place to define a color, as there is a selection tool to visually choose the colours of backgrounds or fonts. This also includes an option to specify RGB values.

Here is an example of how a slide's font color is specified in the JSON editor, under the 'text' tag:

```

▼ English-LCD-HD [2]
  ► 0 {5}
  ▼ 1 {5}
    name : R3
    ▼ text [1]
      ▼ 0 {6}
        value : move\nlove
        horizontal : center
        vertical : center
        fontSize : 150
        fontFamily : Arial
        color : rgb(51,255,153)

```

Within the default slide, the font color is labelled as 'foregroundColor', within the main sub-tags of the 'default' section:


```
▼ defaults {7}
  backGroundColor : rgb(0,245,123)
  foregroundColor : rgb(0,0,163)
```

The backGroundColor is treated the same way in individual slides as in the default slide.

5.2 Positions

This section of the code contains any pre-defined spatial positions which will be used to define where content is placed within a slide. Certain terms are known by the software and can be used directly:

```
center
left
right
top
bottom
```

Each position has a name, a horizontal and a vertical component. Any position other than the pre-defined terms listed above should be entered as a number corresponding to the horizontal or vertical pixel position. A common pre-defined position for NNL paradigms is 'MidPos':

```
▼ position [1]
  ▼ 0 {3}
    name : MidPos
    horizontal : center
    vertical : center
```

Pixel numbering starts on the top left with (horizontal; vertical) co-ordinates of (0,0).

5.3 Conditions

Conditions can be set at the trial or block level. All paradigms must have at least one active condition trial or block. Conditions are specified as 0 (= rest/baseline/control) or a number between 1 and 9 (active). For a paradigm with only 1 type of active condition (e.g. Word Generation), all active conditions are specified as '1'. This information is then saved as part of the design file, to facilitate automated post processing of T-Maps in nordicBrainEx software.

Here is an example of how conditions are set within a trial:

```
▼ blocks [9]
  ► 0 {5}
  ► 1 {5}
  ▼ 2 {5}
    name : Rest2
    ► trials [10]
      condition : 0
```

This example shows a block called 'Rest2' containing 10 trials, and the rest condition has been set at the block level.

6 BOLDSettings

This section allows nordicAktiva to retain compatibility with nordicBrainEx post processing software. Design files saved after running a paradigm (see User Manual) will contain a set of default pre-processing settings, which are defined here, as well as pre-defined contrasts which are necessary for producing T-Maps using the GLM analysis.

Below is an example of how the settings and contrasts can be specified:

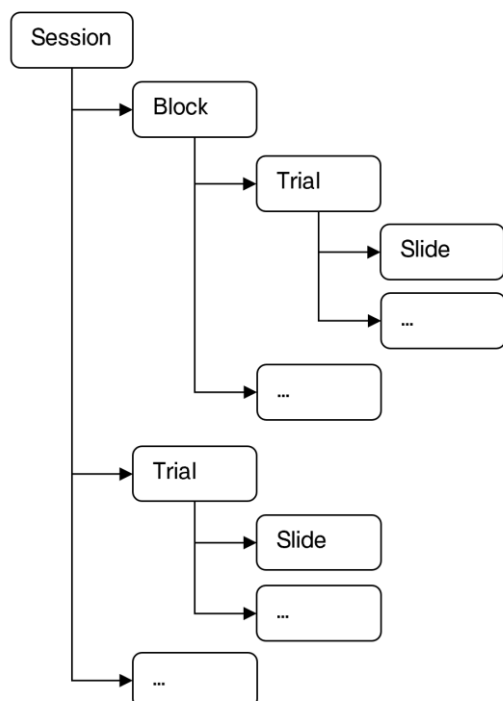
```
▼ BOLDSettings {2}
  ▼ process {6}
    ▼ gaussian {5}
      gaussDim : 3D
      smoothOn : ☒ true
      x_dir : {value}
      y_dir : {value}
      z_dir : {value}
      glmOn : ☒ true
    ▼ motion {1}
      motionOn : ☒ true
      preProcOn : ☒ true
    ▼ sliceTime {4}
      ascendingOn : ☒ true
      interleavedOn : ☐ false
      oddOn : ☐ false
      sliceTimeOn : ☐ false
    ▼ temporal {2}
      cutOff : 180
      temporalOn : ☒ true
  ▼ contrasts {1}
    ▼ contrast {2}
      value : 1
    ▼ attributes {1}
      name : Rhyming ON
```

Further details of the options, and how to define contrasts, can be found in the nordicBrainEx documentation.

Note that the design files can also be created and edited within nordicBrainEx. This just provides a default set of values.

7 Stimuli

The actual stimuli are organized as a hierarchy with a session consisting of blocks and trials. A block consists of one or more trials. A trial consists of slides and registers. The hierarchy is illustrated below:



Sessions and trials are both required for all paradigms. Blocks serve as an optional step in order to organize multiple trials into regular periods of active or rest condition content. Blocks must contain at least one trial, and all trials must contain at least one slide.

7.1 Slides

A slide is essentially a unit of content. Slides are the central element of the paradigm. They define the lowest level of a combination of sounds, pictures, text and videos. Separate sets of slides exist for different languages (where text is used) and different versions (where you may need different sizes of pictures / text, for example, for different display resolutions).

The most straightforward way to define content within a slide is to use the Slide Builder user interface, which allows you to place text, and import content (see User Manual). Colors (background and foreground / font) and positions can be defined at the slide level for individual slides. If not specified, default values will be used.

Below is an example of a slide containing text:

```

▼ slide {3}
  ▼ English-LCD-HD [1]
    ▼ 0 {6}
      name : l1
      ▼ text [1]
        ▼ 0 {6}
          value : yellow
          horizontal : center
          vertical : center
          fontSize : 150
          fontFamily : Arial
          color : orange

```

To specify media content, a file path to the location of the item is specified. Below is an example referring to a picture:

```

▼ VisualSystem_HD [2]
  ▼ 0 {2}
    Picture : ./Pictures/start_VSHD.jpg
    name : StartSign

```

Here there is a slide specifically for a VisualSystem-HD version of a paradigm. The name of the slide is StartSign, and it refers to a picture start_VSHD.jpg, which can be found at the specified location.

The location of the installation of nordicAktiva is specified during the installation process. Within this folder (by default C:/ProgramFiles/nordicAktiva), there is an 'assets' folder, and any paradigms within the library (NNL or User) are listed as individual folders within './assets/paradigms'. The names of the folders must match exactly the names of the paradigms, and it is these folders which act as the default location for the software to look for pictures. For this example, the full file path of the picture might be: C:/ProgramFiles(x86)/nordicAktiva/assets/paradigms/General StartStop_TR2s/Pictures/start_VSHD.jpg.

7.1.1 Expected Response and 'Clear' command

In addition to the content, a slide may have an associated 'Expected Response'. This allows the user to specify whether any button presses would be expected from the subject in response to a given slide, and therefore allows the software to calculate an accuracy of response while the paradigm is running.

Note that this accuracy information is only available during the paradigm run. It is not saved within the log file, due to the many possible variations the user may wish to use to calculate an accuracy measure. For further information on the calculation of response accuracy during the running of paradigms, see the main User Manual.

Adding an expected response to a slide is easiest within the Slide Builder interface (see User Manual). Multiple button presses are permitted from any of the following keys:

- A = Left Thumb (LT)
- B = Left Index finger (LI)
- C = Right Index finger (RI)

D = Right Thumb (RT)

Below is an example of the syntax for an expected response within the JSON Editor:

```
▼ slide {2}
  ▼ English [32]
    ► 0 {2}
    ► 1 {2}
    ▼ 2 {4}
      name : 11
      text : yellow
      color : orange
      ▼ expectedResponse {2}
        ▼ key [2]
          0 : C
          1 : D
```

As you can see, the expected response is programmed as an additional tag along with the other attributes of the slide. The sub-tag 'key' is used to specify that either C or D are expected (e.g. either button on the right hand).

In addition to the 'key' tag, a 'count' tag can also be specified to determine a specific number of button presses to be expected in order to denote a correct response. If this count is set to 0, an unlimited amount of responses are expected, if it has a number between 1 and 99 this dictates the exact amount expected for this slide, and if it is not defined, 1 response is expected. Here is the same expected response shown above, with a count included:

```
▼ expectedResponse {2}
  count : 1
  ▼ key [2]
    0 : C
    1 : D
  clear : ☒ true
```

The additional binary 'clear' command (true or false must be specified) can be used to clear the accuracy calculation after the slide and set it back to zero. This corresponds to the 'clear' check box in the 'Add Expected Response' pop-up within the Slide Builder (see User Manual).

7.2 Trials

A trial makes a slide into a dynamic piece of content. Each trial must contain at least one slide, and a duration (in milliseconds) is assigned to each slide. The condition should be specified at the trial level, if trials are used in isolation to build up the session. For trials which will be inside a block, the condition will be specified at the block level to avoid conflicts.

The syntax of a basic trial is given below:

```

▼ trial [13]
  ► 0 {2}
  ► 1 {3}
  ▼ 2 {3}
    name : R1
    condition : 0
    ▼ show {2}
      item : R1
      duration : $OnShowDuration

```

Each trial has a name, a condition (unless it is within a block, and therefore the condition is set at the block level), and a 'show' command which shows an 'item', corresponding to the name of the slide to be displayed. Each trial must also have a duration. This can be specified in milliseconds, or as in the above example, as a variable name which has been previously defined.

7.2.1 Start trial

Every paradigm must have a 'Start' trial. This is built in to the NNL paradigms and in to the provided templates in the Paradigm Builder, and cannot be deleted from the Paradigm Builder. The 'Start' trial has the name 'Start', which means that name cannot be used elsewhere. It listens for the 's' key from the SyncBox and instructs the software to proceed to the next slide. It is essential for making sure the paradigm can be run in synchronization with the scanner.

The slide specified within the Start trial by default has the name 'Start', but it can be changed. It should contain any instructions or information which you would like the subject to view prior to the start of the paradigm.

7.2.2 Register key

As with the Start trial, other trials can contain instructions to wait for a key to be registered before proceeding. The permitted keys are:

s = sync pulse from SyncBox
a = Left Thumb (LT)
b = Left Index finger (LI)
c = Right Index finger (RI)
d = Right Thumb (RT)

All paradigms provided by NNL use the Start trial ONLY for triggering the run of the paradigm, and subsequent content simply proceeds without waiting for any further keys. The timing accuracy of trials running simply according to the computer clock is considered sufficiently accurate for fMRI experiments.

7.2.3 Abort

If a register key is set, there is an option to add an 'Abort' command, although no user interface field exists for this. The purpose of the Abort command is to stop the stimulus within the current trial (regardless of the duration set). For example, any text on the screen would disappear. 'Abort' requires a true / false value. An example is shown along with the Overlap functionality below.

7.2.4 Overlap

If a trial is aborted in response to a registered key, there is an additional option to immediately proceed to the next slide using an 'Overlap' key.

NOTE: This will bring forward the timings of subsequent content. Please use the log file to determine that all content has been shown at the expected time.

Overlap also requires a true / false value.

In the below example, a trial 'Trial4' shows an item 'Slide7'. If no key press is registered, the slide will show for 5000 ms. If the 'a' key is registered twice (as per the 'number' parameter) during this slide, the content will be removed from the screen using the 'abort' command, and the 'overlap' is used to proceed immediately to the next trial.

```
name : Trial4
▼ show {4}
  item : Slide7
  duration : 5000
  abort : ☒ true
  overlap : ☒ true
condition : 1
▼ register {2}
  key : a
  number : 2
```

7.3 Blocks

The Block defines a combination of Trials. This gives the option of running the same sequence several times in repetition, or running trials in random order. As mentioned, the Condition is set at the block level, if blocks are used.

A "repetitions" tag defines how many times each of the trials of the block shall be repeated. An "order" tag dictates whether the trials will be displayed in a consistent (sequential) or random order. A "1" represents random order, for "0" the order is kept as presented in the list, eg the slides are presented sequentially.

Below is an example:

```
▼ blocks [3]
  ▼ 0 {5}
    name : Block1
    condition : 0
    order : 1
    repetitions : 5
  ▼ trials [4]
    0 : Trial2
    1 : Trial3
    2 : Trial4
    3 : Trial5
```

Here the block named “Block1” is a rest / baseline condition (=0) block. Trials named “Trial2”, “Trial3”, “Trial4” and “Trial5” will be presented randomly (order of 1), and the presentation of the list of trials will be repeated 5 times.

7.4 Session

The Session is what brings the trials and blocks together to make the entire paradigm. ‘runBlock’ and ‘runTrial’ commands are used to run blocks and trials. Within ‘runblock’ and ‘runtrial’, each trial or block to be run is referred to by name, and with an ‘xmlPosition’ tag which dictates where in the order of the paradigm it sits. Below is an example:


```
▼ session {2}
  ▼ runblock [6]
    ▼ 0 {2}
      value : Left
      xmlPosition : 2
    ▼ 1 {2}
      value : Right
      xmlPosition : 3
    ▼ 2 {2}
      value : Left
      xmlPosition : 5
    ▼ 3 {2}
      value : Right
      xmlPosition : 6
    ▼ 4 {2}
      value : Left
      xmlPosition : 8
    ▼ 5 {2}
      value : Right
      xmlPosition : 9
  ▼ runTrial [5]
    ▼ 0 {2}
      value : Start
      xmlPosition : 0
    ▼ 1 {2}
      value : Fixation
      xmlPosition : 1
    ▼ 2 {2}
      value : Fixation
      xmlPosition : 4
    ▼ 3 {2}
      value : Fixation
      xmlPosition : 7
    ▼ 4 {2}
      value : Fixation
      xmlPosition : 10
```

Here there are two blocks, 'Left' and 'Right', and one trial, 'Fixation'. The 'Start' trial has a position of 0, meaning it will be run first, followed by a fixation with position 1, then block 'Left' with a position of 2, and so on.

8 Resources

For any additional assistance with paradigm development / modification, or converting paradigms, there are a number of resources available:

NNL Forum – This is a space where questions can be asked directly to other NNL users and staff, paradigms can be shared, and frequently asked questions can be accessed. Some introductory tutorial videos are also available.

Register for an account for access to the Academy and Forum here:

<https://www.nordicneurolab.com/signup>.

nordicAktiva User Manual – The User Manual for the software provides instructions for the development of paradigms using the Paradigm Builder interface. It can be found within the nordicAktiva Help Menu as a PDF in English. Translated versions, as well as the latest version in English, can be found at the Resources web page for nordicAktiva:

<https://www.nordicneurolab.com/help-all/nordicaktiva>.

Application Support – Our scientific team is available to assist on specific issues via their e-mail address: applicationsupport@nordicneurolab.com. We also very much value your feedback, so please don't hesitate to reach out with any comments and suggestions.