

SeekTake

A Whisker client

by Rudolf Cardinal

www.whiskercontrol.com

Copyright (C) Cambridge University Technical Services Ltd.

Distributed by Campden Instruments Ltd (www.campden-inst.com)



SeekTake

© Cambridge University Technical Services Ltd

All rights reserved. No parts of this work may be reproduced in any form or by any means - graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems - without the written permission of the publisher.

Products that are referred to in this document may be either trademarks and/or registered trademarks of the respective owners. The publisher and the author make no claim to these trademarks.

While every precaution has been taken in the preparation of this document, the publisher and the author assume no responsibility for errors or omissions, or for damages resulting from the use of information contained in this document or from the use of programs and source code that may accompany it. In no event shall the publisher and the author be liable for any loss of profit or any other commercial damage caused or alleged to have been caused directly or indirectly by this document.

Printed: March 2024 in Cambridge, UK

Creator (Whisker)

Rudolf N. Cardinal

Design and Programming (Whisker)

Rudolf N. Cardinal

Michael R. F. Aitken

Legal Advisor (CUTS)

Adjoa D. Tamakloe

Sales (Campden)

Julie Gill

Contacting the authors:

For information about Whisker, visit <http://www.whiskercontrol.com/>.

If you have sales enquiries about Whisker, contact Campden Instruments Ltd at <http://www.campden-inst.com/>.

If you have comments or technical enquiries that cannot be answered by the sales team, contact the authors:

Rudolf Cardinal (rudolf@pobox.com)

Mike Aitken (m.aitken@psychol.cam.ac.uk)

Table of Contents

Foreword	1
Part I SeekTake	2
1 About SeekTake	2
2 Required devices	2
3 Using the task	3
4 Parameters	4
5 Database structure	8
Index	9

Foreword

WARNING

Whisker is a system designed for research purposes only, and should never be used to control medical apparatus or other devices that could endanger human life.

DISCLAIMER

The authors, copyright holders, and distributors disclaim all responsibility for any adverse effects that may occur as a result of a user disregarding the above warning.

1 SeekTake

1.1 About SeekTake

Purpose

Seeking-taking task.

Software requirements

Requires Whisker v2.0 or greater.

Data storage

- Text-based output to disk.
- ODBC data storage to a database (supplied).

Author

Rudolf Cardinal (rudolf@pobox.com).

Copyright

Copyright © Cambridge University Technical Services Ltd

Sample articles using this or similar tasks

- Olmstead MC, Parkinson JA, Miles FJ, Everitt BJ, Dickinson A (2000). Cocaine-seeking by rats: regulation, reinforcement and activation. *Psychopharmacology* **152**: 123–131 [<http://link.springer.de/link/service/journals/00213/papers/0152002/01520123.pdf>]

Version history

- v1.0 - written 26-28 July 2000
- v1.3 - most development here; see `version_tracker`
- v2.0 - 15 Aug 2004
- v2.1-2.3 - minor bug fixes and feature additions (to 22 Nov 2003)
- v2.4 - ability to use the task with only noncontingent CSs (**see [Parameters for a safety warning](#)**), to use tone/houselight as part of the CSs, and XML configuration files.
- v2.5 (11 Nov 2004) - for each response, record whether noncontingent CS on/off (for which, noncontingent CS code rewritten a bit).
- v2.6 (8 March 2007): easier compilation for users.
- v3.0 (12 Jan 2009): Server default changed from "loopback" to "localhost" (Windows Vista compatibility and more general standardization).
- v3.1 (14 Apr 2015): Rebuild for WhiskerClientLib 4.62 with its new socket code.
- v3.2 (14 Mar 2020): New options for degrading contingency.

1.2 Required devices

The program requires to claim devices in groups named **box0**, **box1**, **box2**... with device names as listed below in bold:

```
// Names of lines the program expects to be able to claim
NOSEPOKE           // input
LEFTLEVER        // input
RIGHTLEVER       // input
```

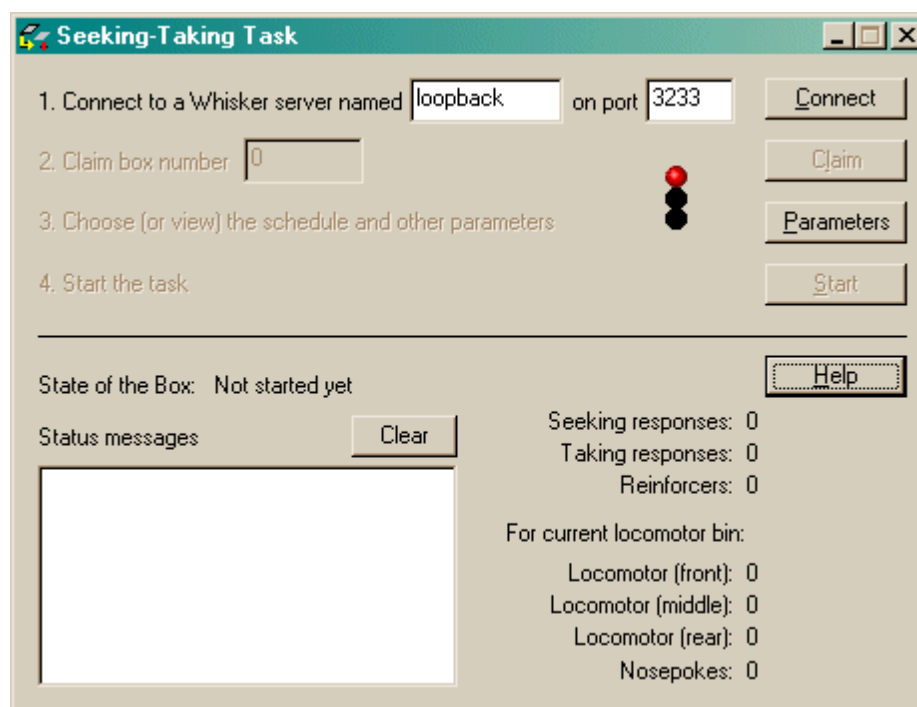
```
LOCOBEAM_FRONT      // input
LOCOBEAM_MIDDLE     // input
LOCOBEAM_REAR       // input
HOUSELIGHT          // output
PUMP                // output
DIPPER              // output
LEFTLEVERCONTROL    // output
RIGHTLEVERCONTROL   // output
LEFTLIGHT           // output
RIGHTLIGHT          // output
CENTRELIGHT         // output
PELLET              // output
TONE                // output

// Aliases used while the program is in full flight, which it therefore expects
not to be present on the server:
SeekingLever
SeekingLeverControl
OtherLight
TakingLever
TakingLeverControl
Stimulus
NoncontingentCS
PIT_CS
```

Please ensure that these devices are available and listed in the device definition file in use by the server.

1.3 Using the task

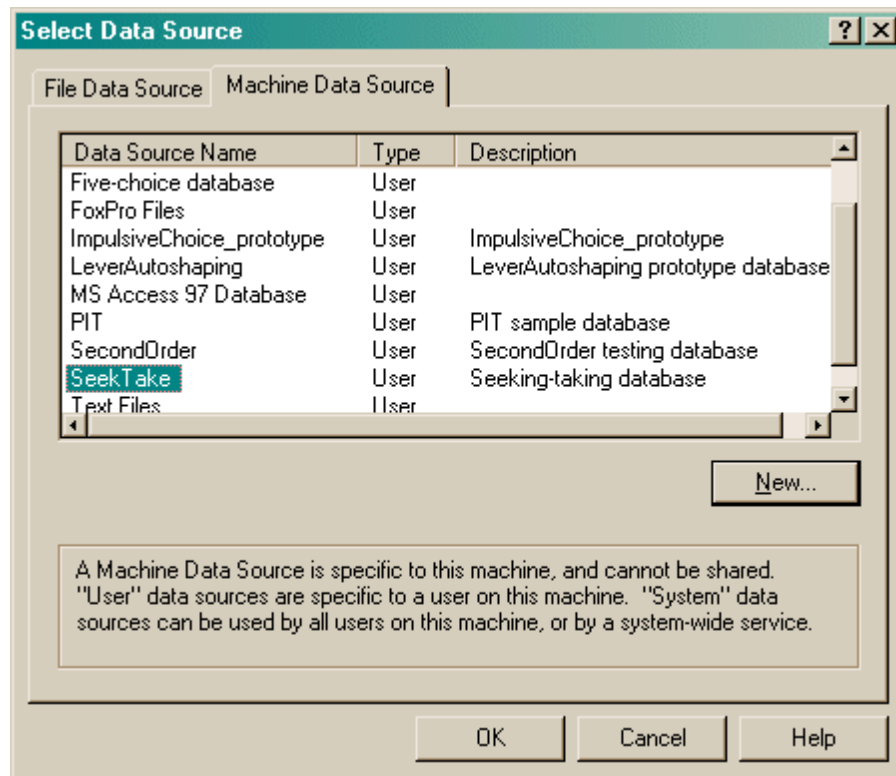
When you run the task, the main screen looks as follows:



You must connect to a Whisker server, claim an operant chamber (box), and set up the [parameters](#)

for your task. Once that's done, the traffic lights will turn amber. When you are ready, press *Start* to begin the task.

When the task finishes, it saves data to disk and pops up a new dialogue box for you to select a database to store the data to. (The data sources are configured under *Control Panel* → *ODBC*.) If you previously specified an ODBC data source in the parameters, that data source is used automatically and you will only see a dialogue box if something goes wrong and the program needs your input.



1.4 Parameters

The task cycles through four basic phases:

1. responding on the **SEEKING** lever (seeking lever present; taking lever absent; houselight on)
2. responding on the **TAKING** lever (seeking lever absent; taking lever present; houselight on)
3. delivery of **REINFORCEMENT** (seeking lever absent; taking lever absent; houselight off; reinforcer-associated stimulus on)
4. **TIMEOUT** (seeking lever absent; taking lever absent; houselight on)

Once one phase is complete, the next begins (unless disabled, in which case it is skipped over).

The schedules for the seeking and taking levers can be varied.

To this basic task, various complex extras can be added:

- A **PRIMING** event can begin one or more cycles (think of this as phase 0). This priming event can include a priming/noncontingent stimulus (possibly different from the reinforcer-associated stimulus), noncontingent reinforcement, and/or presentation of the taking lever (with or without the requirement to respond, and with or without the option to reinforce the response either with a stimulus or primary reinforcement). The dialogue box below details all these options more clearly.

- The seeking phase can be accompanied by presentation of a third type of stimulus, known as the **PIT (Pavlovian-instrumental transfer)** stimulus, on some or all seeking cycles.
- **Noncontingent stimuli** can be presented, desynchronized with the main seeking/taking cycle. Exceptionally, the task can be run with no levers but still presenting these noncontingent stimuli, which makes the task completely noncontingent (Pavlovian). **Note: see below for a safety warning if you allow the task to give reinforcement with no levers present (i.e. noncontingent reinforcement).**

The parameters dialogue box looks like this:

Set parameters for the Seeking-Taking Task

Subject details: Load config, Rat ID: xxx, Session number: 1, Save config, Comment: <generated automatically>

Quick schedule config: FR1-pump, Seek-Take, Take-Take, Seek-Seek, OK, Cancel

Data recording: Set summary, Set response, ODBC data source name (see Control Panel). Blank to choose later: [], Pick

DEVICES

Lever assignment: Seeking LEFT, taking right Seeking RIGHT, taking left

Reinforcer-associated CS: Light over taking lever Centre light Light over seeking lever (abnormal)
 Tone Houselight (NB also used by main task)

Priming/noncontingent stimulus: Light over taking lever Centre light Light over seeking lever (abnormal)
 Tone Houselight (NB also used by main task)

Reinforcer: Pump - Infusion duration (s): 7.28
 Dipper - # dips: 1, Dip time (s): 5, Inter-dip time (s): 1, Down at rest
 Pellets - # pellets: 1, Pulse length (ms): 45, Time between pellets (s): 0.5

SCHEDULE

PRIMING event Prime all cycles Prime randomized in pairs Prime alternate cycles First cycle IS primed

Completely NONCONTINGENT stimulus, comprising: Priming CS Taking lever Reinforcement (as above)
Stimulus duration (s) (N.B. reinf. duration as above): 7, Time from START of stim. to START of next phase (s): 10

Response on taking lever required. Present CS with lever Reinforce press
 Present CS after response, for 0 s. Wait 10 s after response before starting next phase.

SEEKING lever FR VR FI RI Ext-Timeout FR1-Timeout Parameter: 30 s, 0

PIT PIT Noncontingent reinforcement during seeking NRDS

TAKING lever FR VR FI RI Ext-Timeout FR1-Timeout Parameter: 1, 0

REINFORCER, accompanied by CS (see above), and with the houselight off.

TIME OUT ... of 6 minutes before the cycle restarts This task is over-complex. :-)

Session time limit (min) (0 for no limit): 180 Maximum # reinforcers (0 for no limit): 20

Locomotor recording bin size (min): 10 Give desynchronized noncontingent CSs Settings for desync. CS presentation

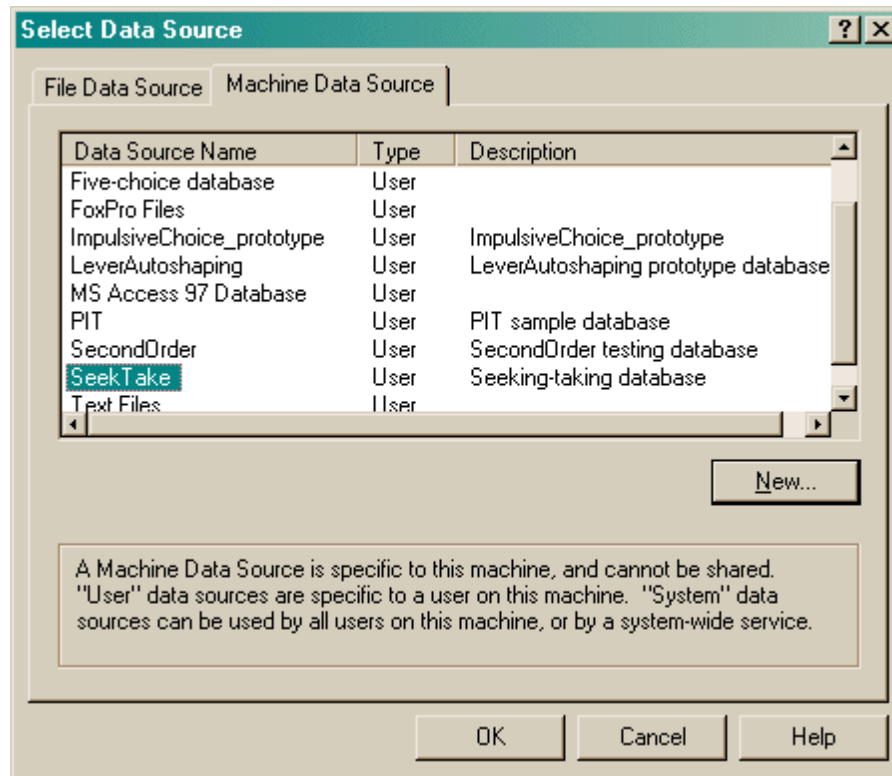
Be careful if you use the houselight as part of a CS; this is not necessarily advised, because it is possible that (for example) a noncontingent CS finishes during a cycle and switches the houselight off as it finishes, leaving the task in the Seeking state but with the houselight off (which is not normal).

Click one of the four "Quick schedule config" buttons to set up standard schedule configurations, or edit the schedule directly.

The **Ext-Timeout** and **FR1-Timeout** are extinction and FR1 schedules, respectively, with the

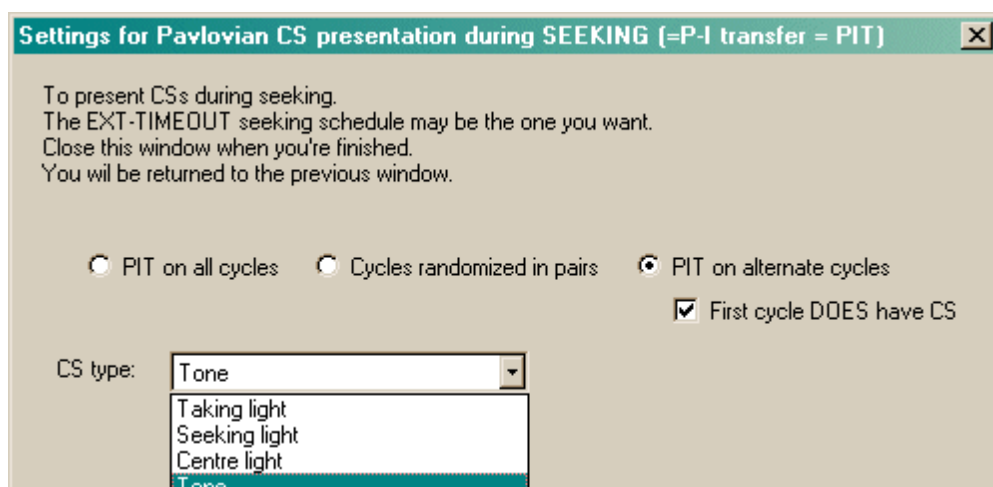
additional feature that the schedule automatically terminates after a specified time (at which point, the next enabled "component" [e.g. taking lever, reinforcer, timeout] takes over.)

To pick an ODBC database **in advance** of finishing, click *Pick* and you will be offered the ODBC Data Source picker (below). Your choice will be recorded and will apply to this subject from now on (or until you specify a different source).



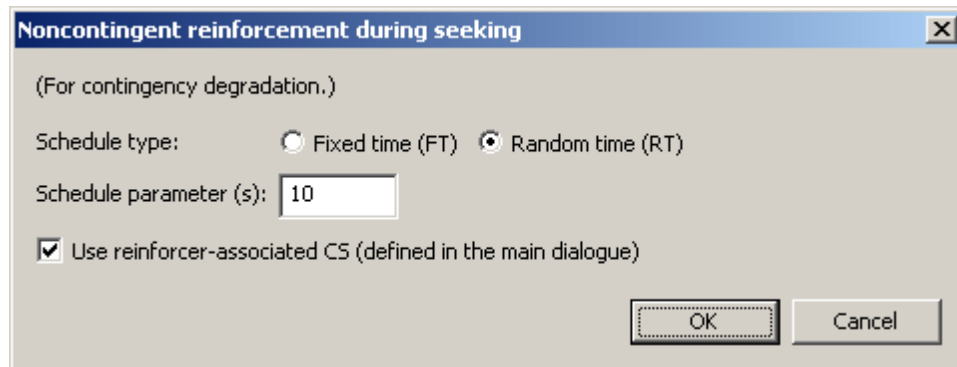
If you don't specify an ODBC data source now, or you delete the value in the "ODBC data source name" box, you'll be asked to choose when the task ends (and that choice will only apply to the session in progress).

To present **noncontingent CSs during seeking (Pavlovian-instrumental transfer, or PIT)**, enable the tick-box labelled "PIT" in the "Seeking" line and click the "PIT" button to set options:

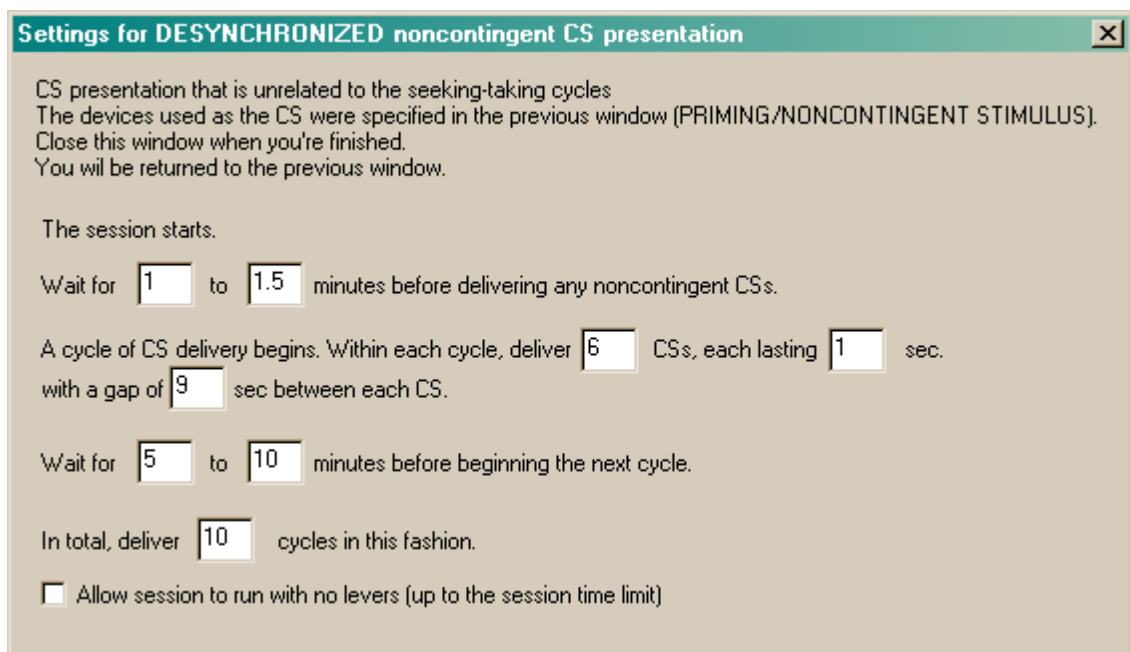


To give **noncontingent reinforcement during seeking** (for contingency degradation), enable the

tick-box labelled "Noncontingent reinforcement during seeking" and click the "NRDS" button to set options:



To give **desynchronized noncontingent CSs independent of the seeking/taking cycle**, enable "Give desynchronized noncontingent CSs" and click "Settings for desync. CS presentation":

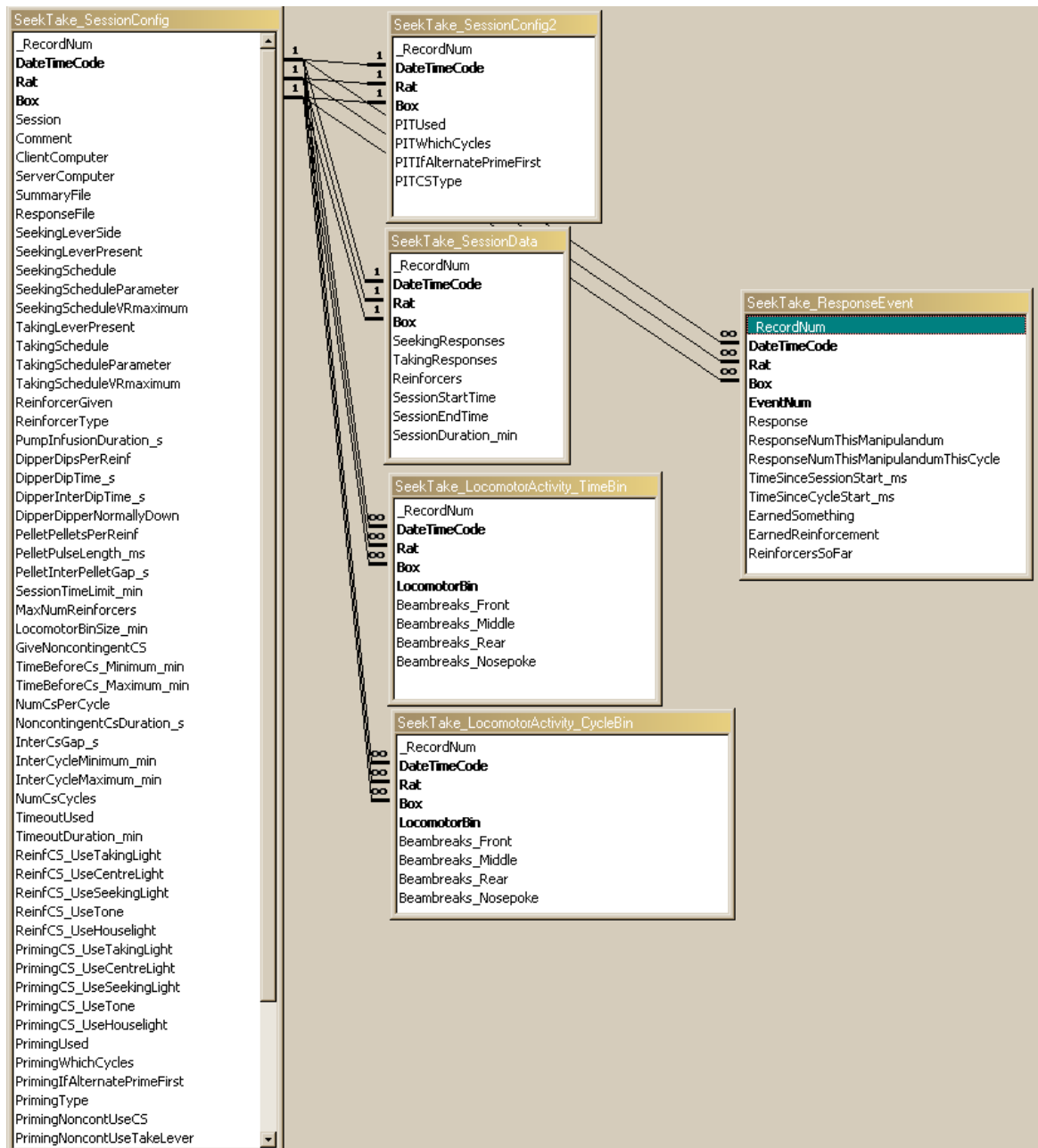


The final option in this dialogue box allows you to run without the seeking or taking levers - allowing a predefined session length and the presentation of noncontingent CSs only. Note that **it is imperative that you check the reinforcement parameters** in this situation. For example, setting this combination - seeking OFF, taking OFF [a combination ordinarily disallowed but allowed if you are giving desynchronized noncontingent CSs and have ticked the "allow no levers" option], **reinforcement ON**, timeout ON with a short timeout - would deliver reinforcer after reinforcer (since the task cycles through: skip seeking -> skip taking -> give reinforcement -> timeout -> restart cycle again). **The program disallows "seeking OFF, taking OFF, reinforcement ON" unless a timeout is specified, but short timeouts would still be dangerous if IV reinforcement is used.**

In general, if you tick the "allow session to run with no levers" box, you must check very carefully what happens - although well defined, the program's behaviour may be counter-intuitive. Caveat emptor.

1.5 Database structure

This is the structure of the SeekTake database:



Index

- S -

SeekTake

- database structure 8
- parameters 4
- required devices 2
- using 3