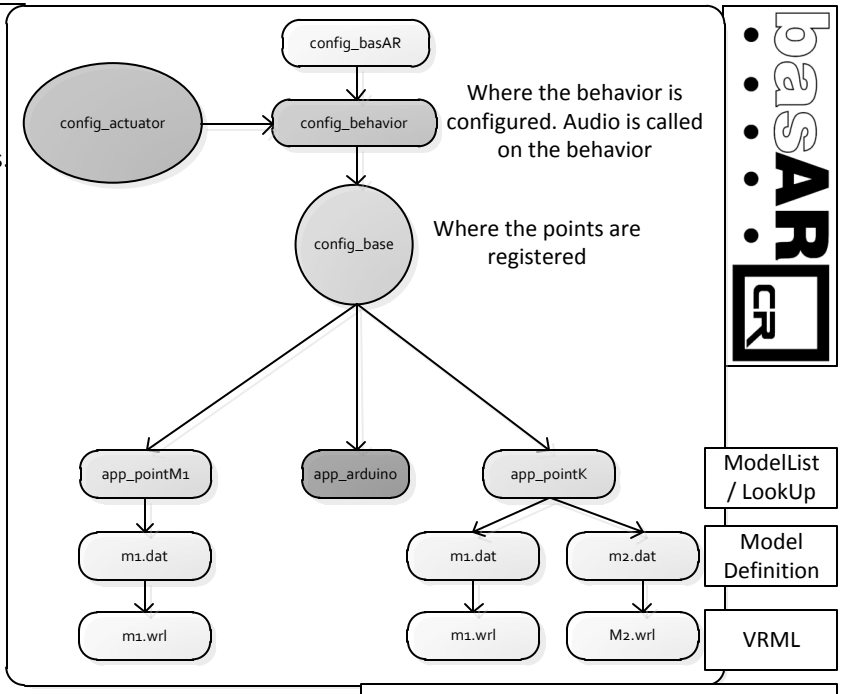
COMMANDS

[PointID] **STAT** [ShowMode] <AUDIO> <OVER?>	Static, no action defined.
[PointID] **DRGF** [ShowMode] <AUDIO> <OVER?>	Drag freely.
[PointID] **DRGRP** [ShowMode] [NextState] <AUDIO> <OVER?>	Drag freely, repels others.
[PointID] **ATTO** [PointWaited] [ShowMode] [NextState] <AUDIO> <OVER?>	Attracts one.
[PointID] **ATTRP** [PointWaited] [ShowMode] [NextState] <AUDIO> <OVER?>	Attracts one, repels others.
[PointID] **ATTA** [ShowMode] [NextState] <AUDIO> <OVER?>	Attracts all.
[PointID] **DRPO** [PointWaited] [ShowMode] [NextState] <AUDIO> <OVER?>	Drop one.
[PointID] **DRPA** [ShowMode] [NextState] <AUDIO> <OVER?>	Drop all.
[PointID] **RPLO** [PointWaited] [ShowMode] [NextState] <AUDIO> <OVER?>	Repels one.
[PointID] **RPLA** [ShowMode] [NextState] <AUDIO> <OVER?>	Repels all.
[PointID] **CHGST** [ShowMode] [NextState] <AUDIO> <OVER?>	Change state.
[PointID] **TRA** [X] [Y] [Z] <Time> <Step>	Translate.
[PointID] **ROT** [X] [Y] [Z] <Time> <Step>	Rotate.
[PointID] **SCL** [X] [Y] [Z] <Time> <Step>	Scale.
[PointID] **CHGM** [ModelToChange]	Change model.
[PointID] **CHGNM**	Change next model.
[PointID] **SETS**	Save actual on start.
[PointID] **SETL**	Save actual on last.
[PointID] **GETS**	Get start on actual.
[PointID] **GETL**	Get last on actual.
[PointID] **ESND** [MSG] <AUDIO> <OVER?>	Send to external.
[PointID] **ERCV** [MSG] [NextState] <AUDIO> <OVER?>	Receive from external
[PointID] **LOADA** [Value]	A <= Value
[PointID] **LOADB** [Value]	B <= Value
[PointID] **ADDA** [Value]	A <= A + Value
[PointID] **ADDB** [Value]	B <= B + Value
[PointID] **AMB**	A <= A – B
[PointID] **BMA**	B <= B – A
[PointID] **NEGA**	A <= -A
[PointID] **NEGB**	B <= -B
[PointID] **MULA** [Value]	A <= A*Value
[PointID] **MULB** [Value]	B <= B*Value
[PointID] SWAB	A ⇔ B
[PointID] **CMP** [GREATER/LESSER/EQUALS] [NEXTSTATE]	If A is (Greater, lesser or equals) than B do next state.
[PointID] **CMPV** [Value] [GREATER/LESSER/EQUALS] [NEXTSTATE]	If A is (Greater, lesser or equals) than Value do next state.



OBS.:
First State must configure all the virtual points.

State change priority

CMP/ CMPV functions
Configuration State
Commands

STATE TYPES

Normal State
BEGIN_STATE [Number]
Actions
END_STATE

Configuration State
BEGIN_STATE [Number]
Actions
END_STATE GO_TO [NextState]

Configuration State with Time
BEGIN_STATE [Number]
Actions
END_STATE GO_TO [NextState] AFTER [Time]

SHOW MODE

HIDE Hide active model and action point
ONLY_BALL Show only the action point
ONLY_OBJECT Show only the active model
BOTHShow active model and action point
FLASH_BALL Show only a flashing action point
SENSE_PROX Shows only the action point with proximity sensor function.

- ATTA and DRPA, (Away and correct)
- RPLA, (Away and Wrong)
- ATTO, ATTRP, DRPO and RPLO (Away, correct and wrong)

ALL_OBJECTSShow all models and action point

BASE POINT DEFINITION

PointName
ActionModel (DEFAULT_IPOINT / EXTERNAL_IPOINT / UserDefined)
ObjectModels (NO_OBJECT / ModelList / LookupTable)
Start Translaction
StartRotation
StartScale
ActionRadius

BASAR CONFIGURATION FILE

```
BASAR – Behavioral Authoring System for Augmented Reality
WINDOWED

VRML wrl/action/ballBlue.dat
VRML Wrl/Action/ballGreen.dat
VRML Wrl/Action/ballRED.dat
VRML Wrl/action/tampa.dat
Audio/explosion.wav 0.5

Audio/backTrack.mp3 LOOP  0.3
Audio/bell.wav      ONCE  0.5

Data/config_behavior

1
Data/config_base

1

ARTKSM Data/config_actuator
```

BASE CONFIGURATION

```
BASE1

ARTKSM
Data/Markers/base.patt
74.0
0.0 0.0
USE_DEFAULT
Audio/bell.wav      ONCE  0.5
Audio/explosion.wav 0.5

2

Pen
DEFAULT_IPOINT
Data/config_pointA
-100.0 -100.0 0.0
15.0 0.0 0.0
1 1 1
900.0

ARDUINO
EXTERNAL_IPOINT
Data/app_arduino
0 0 0
0 0 0
0 0 0
0
```

ACTUATOR CONFIGURATION FILE

```
ARTKSM1

Data/Markers/pa.patt

37.0
0.0 0.0
NO_COVER

VRML wrl/Action/pa.dat
DEFAULT_IPOINT
20.0 0.0 0.0
400.0
```

BEHAVIOR CONFIGURATION FILE

```
BEGIN_STATE 1
        1 STAT ONLY_BALL Audio/explosion.wav
        2 DRGF ONLY_OBJECT
        3 DRGF ONLY_OBJECT
        4 ATTO 2 ONLY_BALL 2 Audio/bell.wav
END_STATE

BEGIN_STATE 2
        2 SCL 2 2 2
        3 SCL 2 2 2
END_STATE  GO_TO 4

BEGIN_STATE 4
        2 STAT ONLY_OBJECT
        3 STAT ONLY_OBJECT

END_STATE
```

ARDUINO EXAMPLE

```
int ledPin = 12;   // select the pin for the LED
int boardled = 13;   // select the pin for the LED
int buttonPin = 2;

int buttonState = 0;
int ledState = 0
int b1Hold = 0;
int val = 0;       // variable to store the data from the serial port
int intNSdata = 0;


void setup() {
  Serial.begin(9600);        // connect to the serial port
}

void loop () {
  // read the serial port
  val = Serial.read();
  buttonState = digitalRead(buttonPin);

              ... // Code to adapt interruption


  switch(val){
  case 1: // CHECK IF ITS aliveTest
    {
      Serial.print(2);        // send back aliveAnswer
      break;
    }
  case 3: // CHECK IF ITS intRequest
    {
      Serial.print(intNSdata);
      break;
    }
  case 4: // CHECK IF Its readRequest
    {
      break;
    }
  default:
    break;
  }

}
```

ARDUINO LOOKUP TABLE FILE

```
COM4 # ARDUINO COM

# ARDUINO LOOKUP TABLE
# THE NUMBERS: 10-254
# TABLE FORMAT:   RQ_N RQ_NAME <NextState>

# CONFIGURATION

1 aliveTest
2 aliveAnswer
3 intRequest
4 readRequest
5 readRequestAnswer

# USER COMMANDS

# DEVICE 1 - Light
10          lightOFF
11          lightON

# DEVICE 2 - Button


20          buttonPressed       5
```

POINT MODEL LIST FILE

```
1                   #NUMBER

MODEL3D VRML Wrl/pen.dat
```

VRML CALLER FILE

```
pen1.wrl            #VRML
0.0 0.0 00.0                #TRANS
0.0 0.0 0.0                 #ROT
0.25 0.25 0.25              #SCL
```

basAR CR