



# Math.IA

## Curso basAR

Christopher Shneider Cerqueira – [christophercerqueira@gmail.com](mailto:christophercerqueira@gmail.com)

Claudio Kirner – [ckirner@gmail.com](mailto:ckirner@gmail.com)

<https://sites.google.com/site/christophercerqueira/projetos/ear/basar>

<http://www.ckirner.com/basar>

# Requisitos

- Windows
- basAR
  - Download:
    - <https://sites.google.com/site/christophercerqueira/projetos/ear/basar>
    - <http://www.ckirner.com/basar>
- Alguns modelos 3D em VRML
- Editor de texto simples, como o Bloco de Notas.

# Roteiro

- Introdução: Metas
- basAR
- Criando infraestrutura, estrutura e conteúdo
- Criando comportamento com controle matemático.
  - Conceitos:
    - Mudança de estado
    - Lógicas de controle.
  - Comandos
  - Atividade

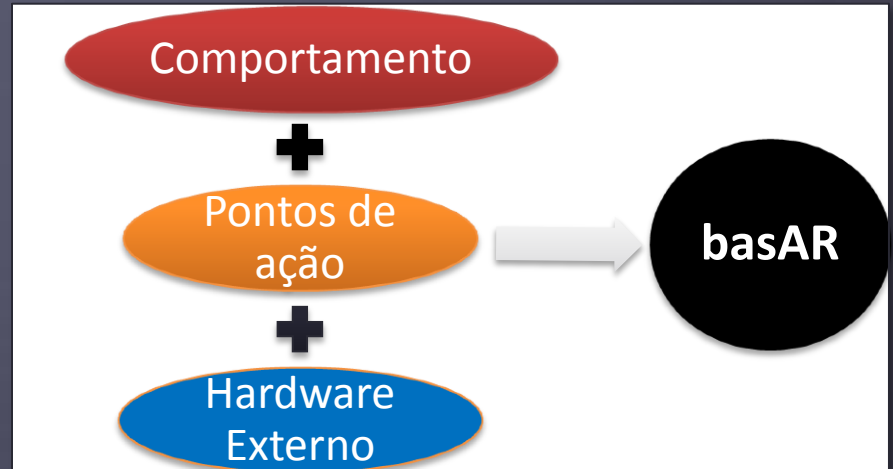
# Introdução

- Criar um cenário de Realidade Aumentada.
  - Dois pontos de seleção e um ponto de exibição
- Utilizar o software basAR.
- Entender a construção do cenário.
- Entender os conceitos de mudança de estado e mudança de atributos

# basAR

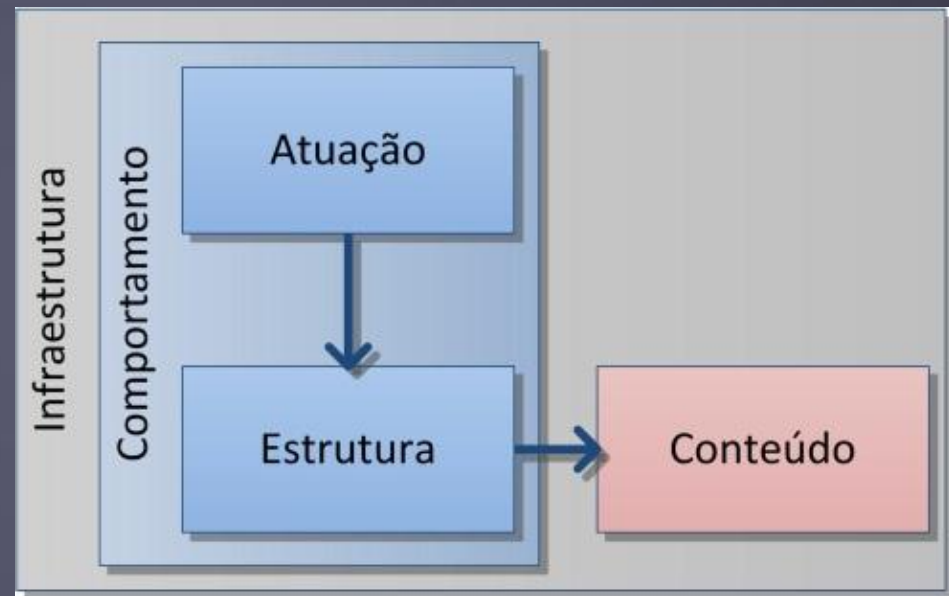
## basAR: Behavioral Authoring System for Augmented Reality

- Meta-ferramenta de autoria de ambientes inteligentes de realidade aumentada .
  - Estrutura de comportamento baseada em pontos de ação
  - Cross-Reality
- COMPORTAMENTO DINÂMICO entre PONTOS DE AÇÃO

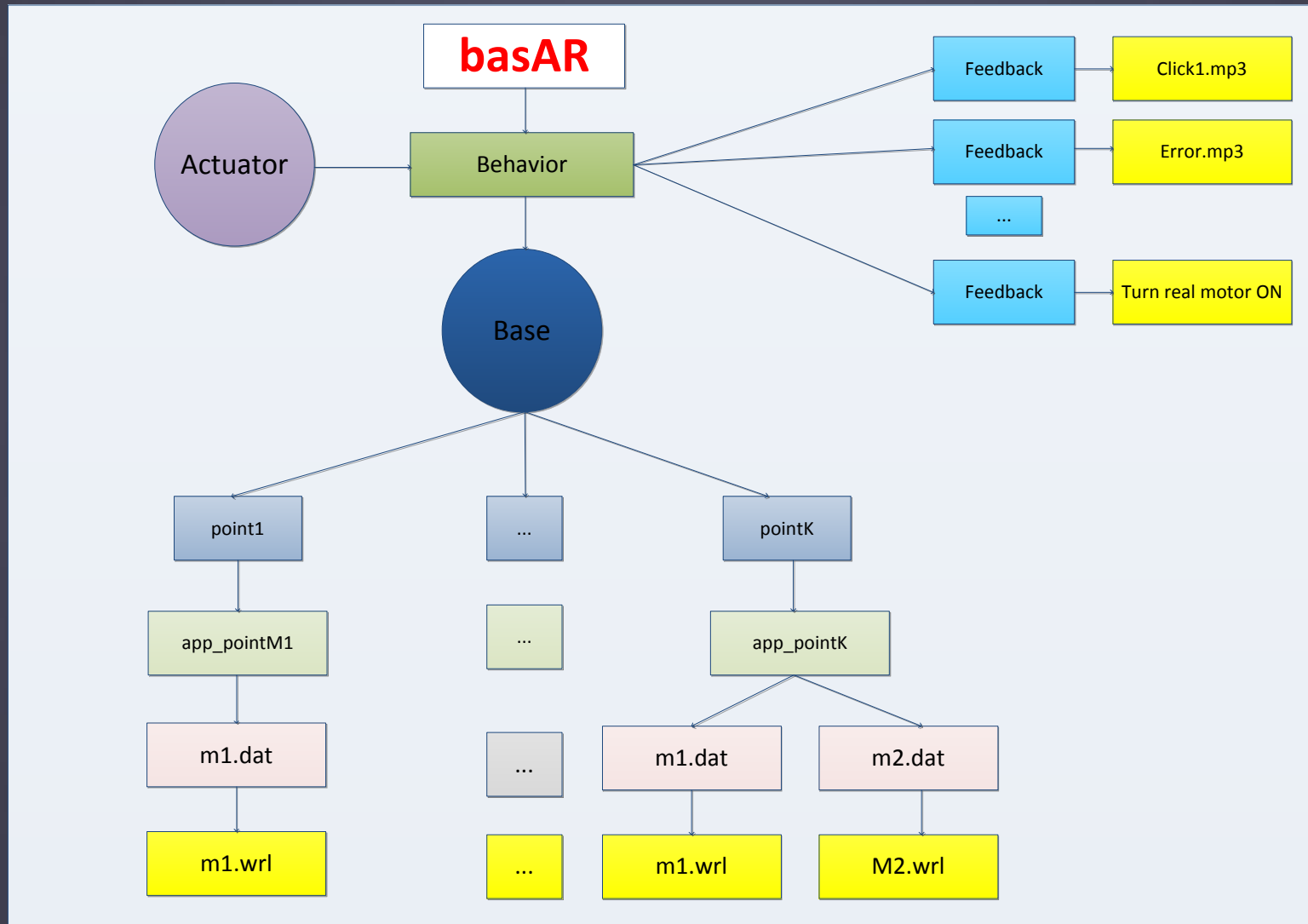


# Aplicação em camadas

- **Infraestrutura:** especificar a área de trabalho (*workspace*) da aplicação
- **Estrutura:** os posicionamentos dos pontos de ação sobre a infraestrutura.
- **Contexto:** modelos, sons e outros objetos definidos na estrutura.
- **Atuação:** método de interação com a estrutura.
- **Comportamento:** regras de interação entre os atuadores e a estrutura.

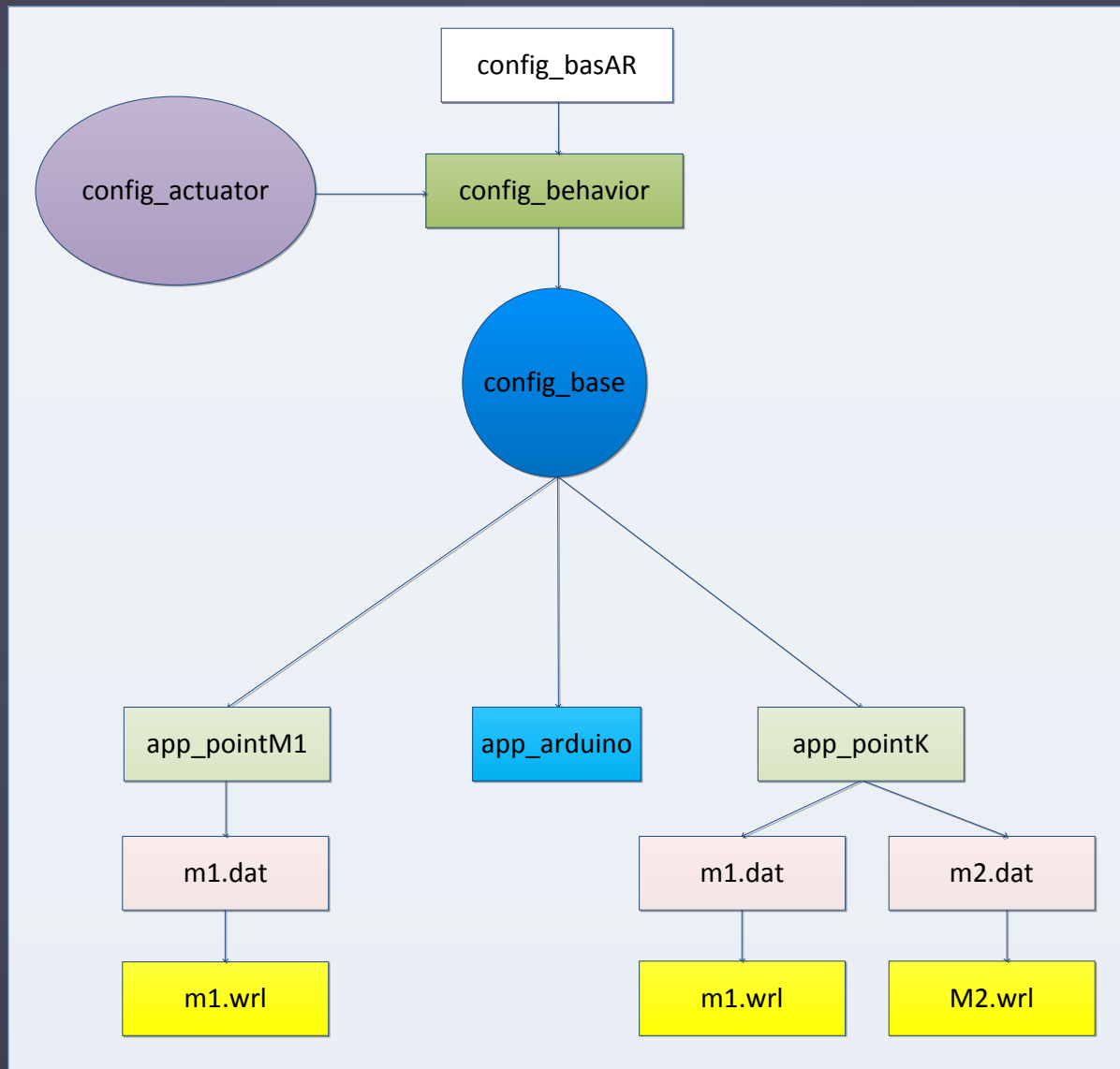


# Hierarquia de objetos





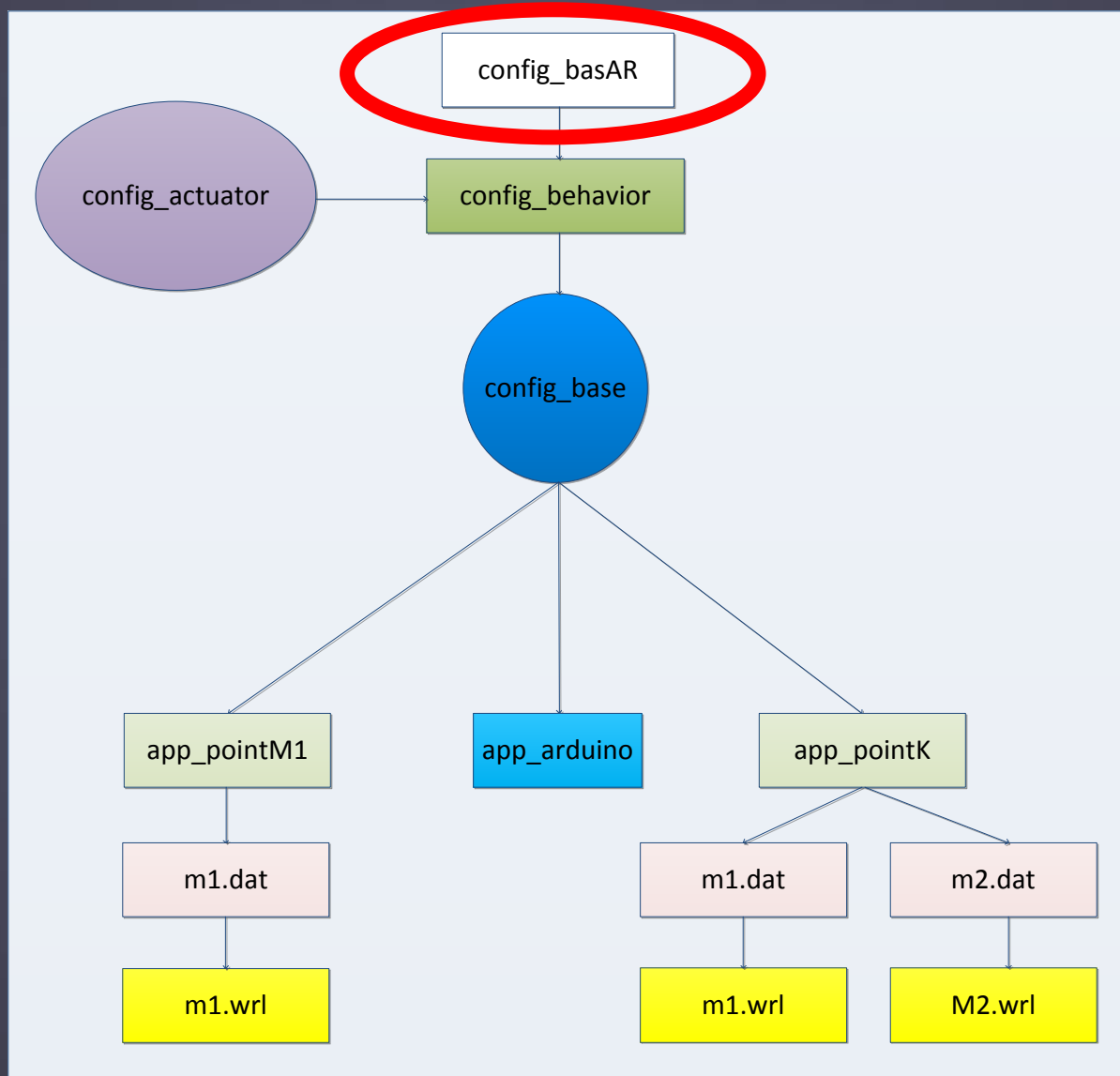
# Hierarquia de arquivos



Criando

# INFRAESTRUTURA, ESTRUTURA, CONTEÚDO E ATUAÇÃO

# Sistema



# Sistema

- Configura:
  - a primeira interligação dos objetos;
  - modo de exibição da janela;
  - objetos padrões.
- Arquivo:  
{basAR}/Data/config\_basAR

## #config\_basAR

basAR

## 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 **# Rule Machine**

## # Bases

1

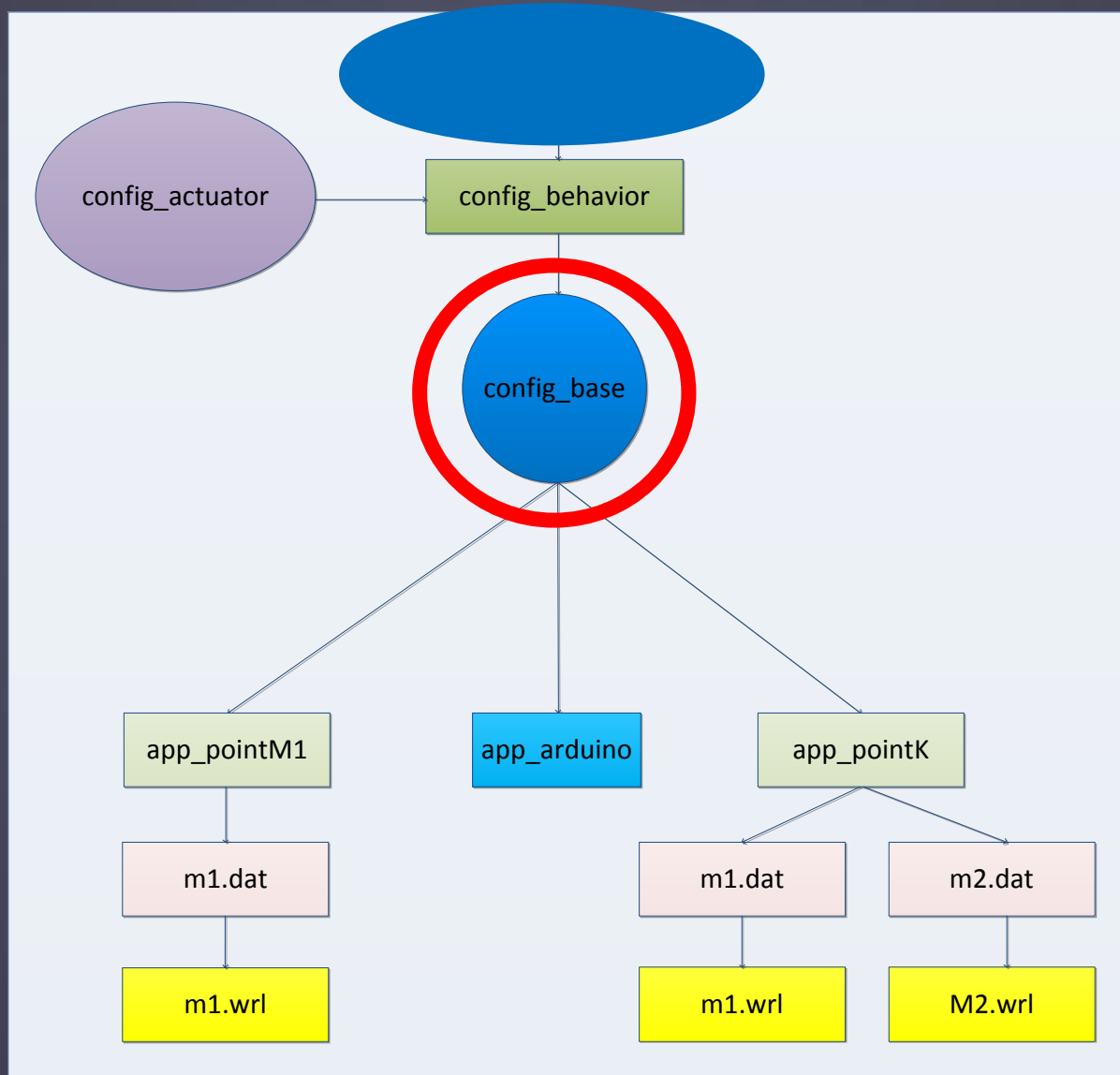
Data/config\_base1

## # Actuators

1

ARTKSM Data/config\_transport

# Infraestrutura



# Infraestrutura

- correlação entre o mundo real e o mundo virtual
- Definição de base
- Arquivo:  
{basAR}/Data/config\_base

```
#config_base1
```

```
# This file contains the setup for a workspace  
BASE1
```

```
# Single ARToolKit Marker configuration
```

```
ARTKSM
```

```
Data/Markers/base.patt
```

```
53.0
```

```
0.0 0.0
```

```
USE_DEFAULT
```

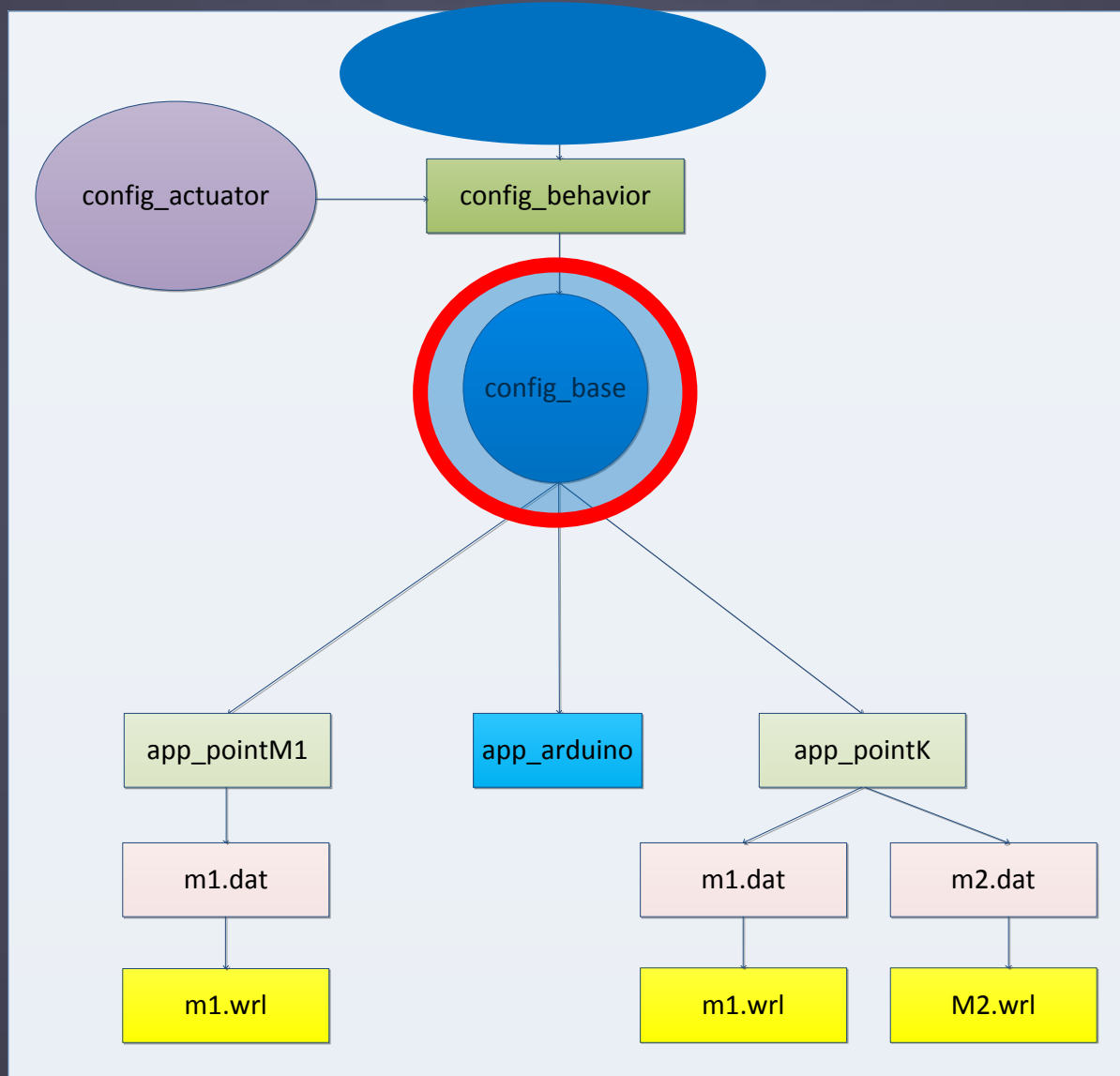
```
# Workspace Sounds
```

```
Audio/bell.wav    ONCE 0.5    # Visible Sound
```

```
Audio/explosion.wav 0.5      # Error sound
```

```
wrl/action/status.dat
```

# Estrutura



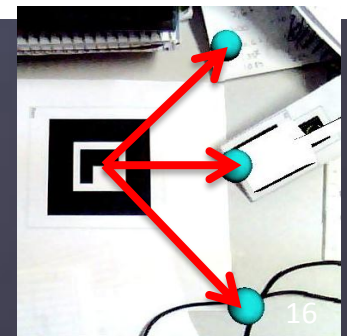
# Estrutura

- Pontos do espaço virtual com **esferas de ação**, **modelos associados** e **comportamento**.
- Localização, orientação e escala dos objetos virtuais.
- Arquivo:  
{basAR}/Data/config\_base

...

Pen	# Point Name
DEFAULT_IPOINT	# Action Model File
Data/app_pen	# OBJECT Model File
20.0 20.0 0.0	# Translation
0.0 0.0 0.0	# Rotation
1.0 1.0 1.0	# Scale
900.0	# Action radius

...





# Exemplo com 3 pontos

...

3

Ponto1

DEFAULT\_IPOINT

**NO\_OBJECT**

100.0 100.0 25.0

0.0 0.0 0.0

1.0 1.0 1.0

300.0

Ponto2

DEFAULT\_IPOINT

**NO\_OBJECT**

100.0 -100.0 25.0

0.0 0.0 0.0

1.0 1.0 1.0

300.0

pontoVisual

DEFAULT\_IPOINT

**Data/App/app\_obj**

100.0 0.0 25.0

0.0 0.0 0.0

1.0 1.0 1.0

0

# Arquivo config\_base completo \_p1

BASE1

ARTKSM

Data/Markers/base.patt

53.0

0.0 0.0

USE\_DEFAULT

Audio/bell.wav    ONCE 0.5

Audio/explosion.wav 0.5

wrl/action/status.dat

3

Ponto1

DEFAULT\_IPOINT

NO\_OBJECT

100.0 100.0 25.0

0.0 0.0 0.0

1.0 1.0 1.0

300.0

# Arquivo config\_base completo \_p2

Ponto2

DEFAULT\_IPOINT

NO\_OBJECT

100.0 -100.0 25.0

0.0 0.0 0.0

1.0 1.0 1.0

300.0

pontoVisual

DEFAULT\_IPOINT

Data/App/app\_obj

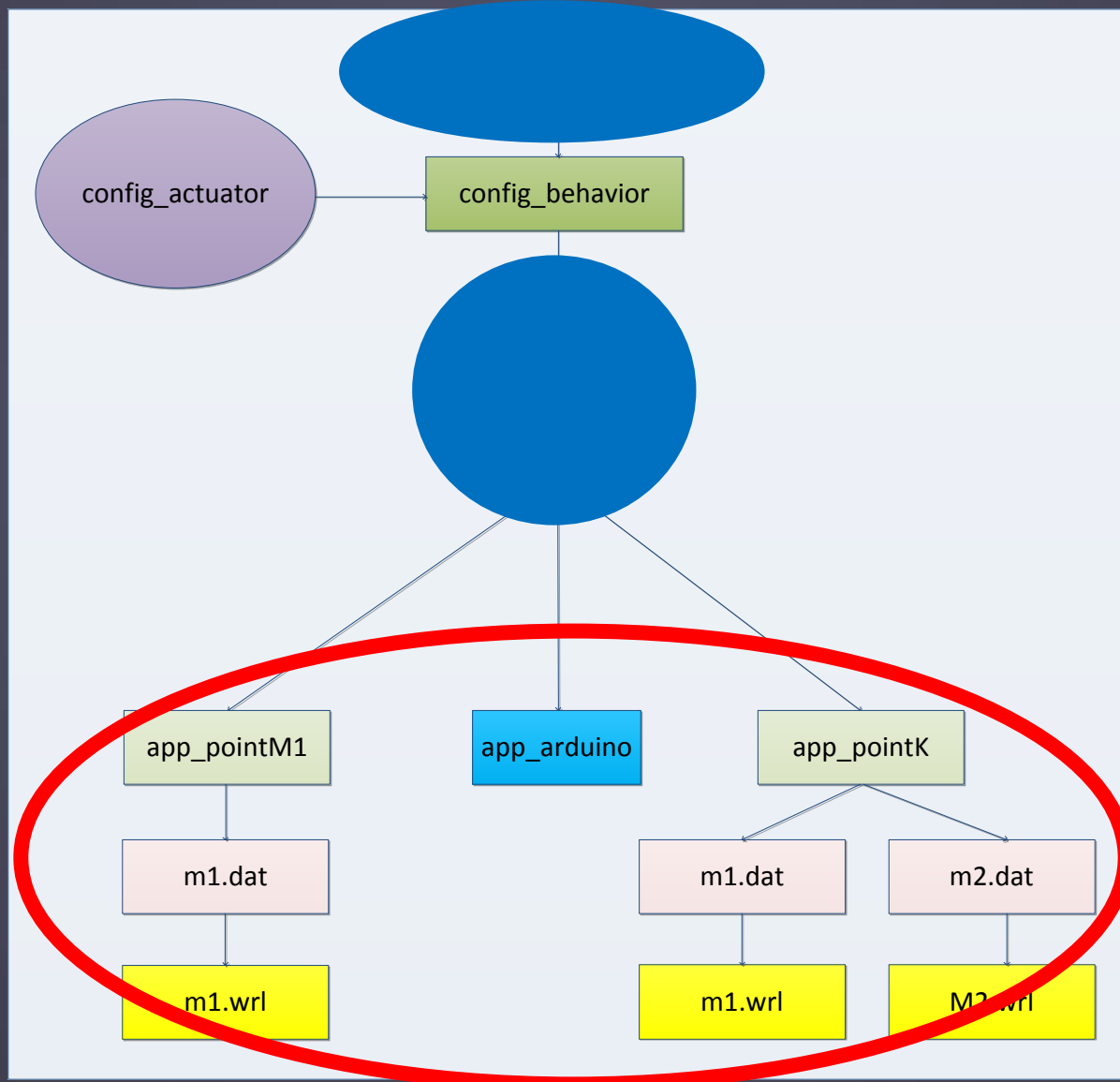
100.0 0.0 25.0

0.0 0.0 0.0

1.0 1.0 1.0

0

# Conteúdo



# Conteúdo

- objetos 3D e áudios que são utilizados.

- Arquivos:

{basAR}/Data/App/app\_obj

{basAR}/Wrl/quad.dat

{basAR}/Wrl/triang.dat

{basAR}/Wrl/quad.wrl

{basAR}/Wrl/triang.wrl

**#app\_obj**

2

**# Number of objects**

MODEL3D VRML Wrl/quad.dat

MODEL3D VRML Wrl/triang.dat

**#quad.dat**

quad.wrl

0 0 0

0 90 0

0 0 0

**# triang.dat**

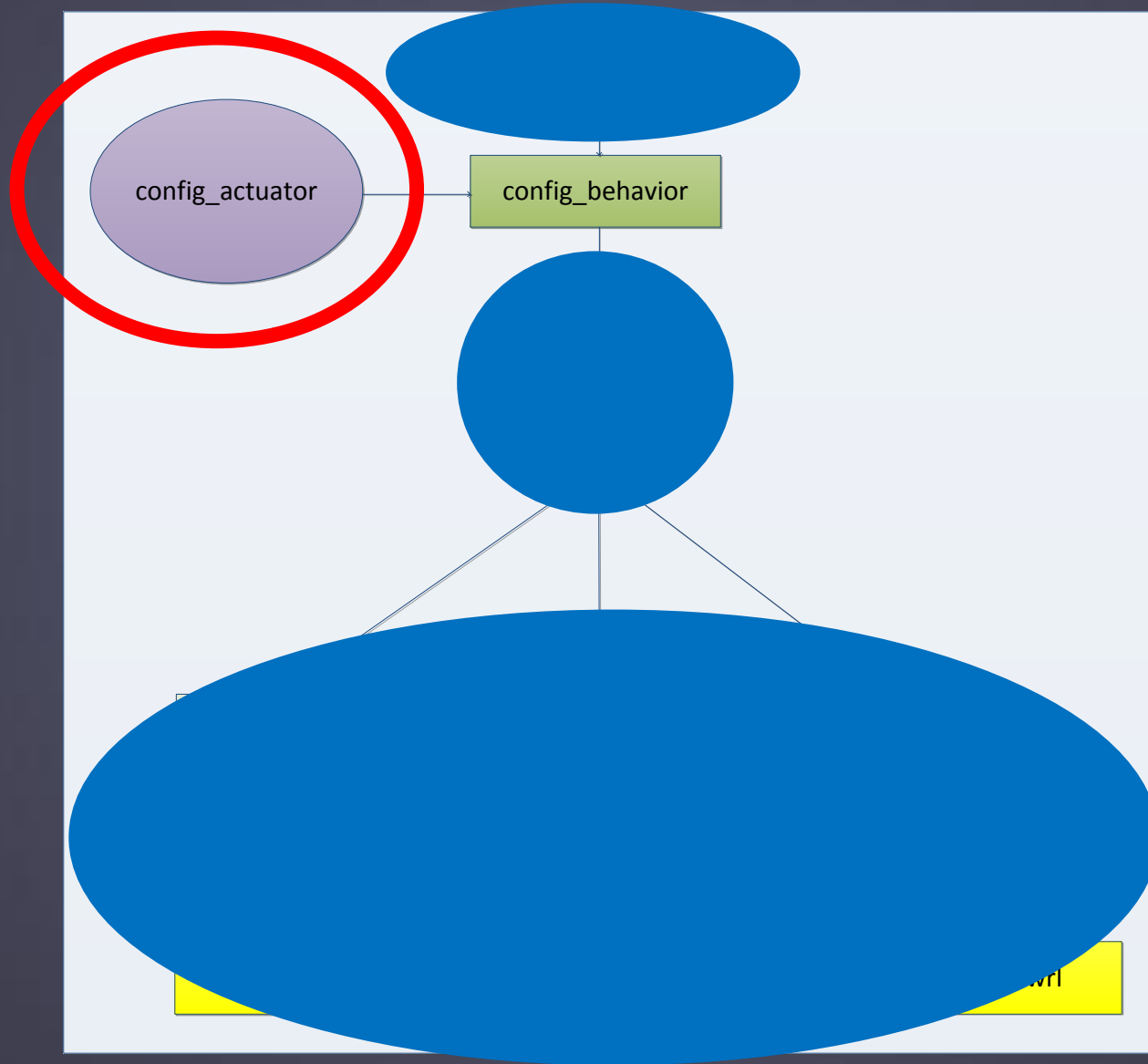
triang.wrl

0 0 0

0 90 0

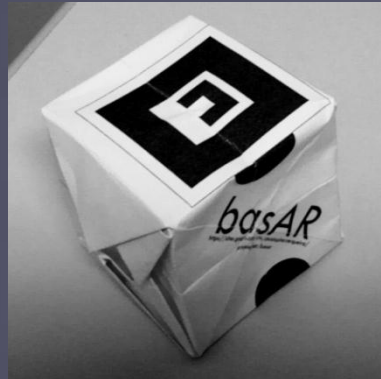
0 0 0

# Atuação



# Atuação

- Interação do usuário com o sistema



## #config\_tranport

ARTKSM1

## # Single ARToolKit Marker configuration

Data/Markers/shovell.patt

37.0

0.0 0.0

USE\_DEFAULT

VRML wrl/Action/ssd.dat

model

# Marker

# Width(mm)

# Central

# Marker cover

# Symbolic

## # Collision point

DEFAULT\_IPOINT

20.0 0.0 0.0

400.0

# Point model

# Translation (x,y,z) (mm)

# Action radius of the point

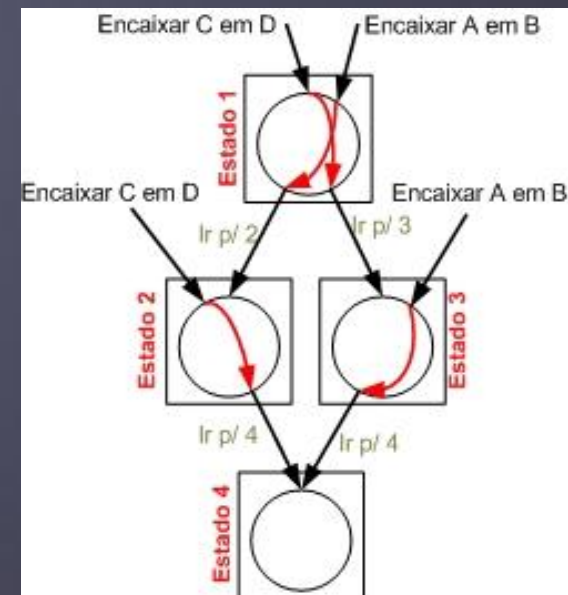
Criando

# COMPORTAMENTO DE CLICKS



# Comportamento?

- Controle do feedback do sistema ao receber estímulo do usuário.
- Baseado em máquina de estados (grafo de cena).
- Ações → Comandos
- Estados → Blocos de comandos



# Exemplo de estado

- Inicia estado
- Ponto 1 estático, exibe tudo
- Ponto 1 estático, exibe tudo
- Ponto 1 estático, exibe tudo
- Finaliza estado

**BEGIN\_STATE 1**

**1 STAT BOTH**

**2 STAT BOTH**

**3 STAT BOTH**

**END\_STATE**

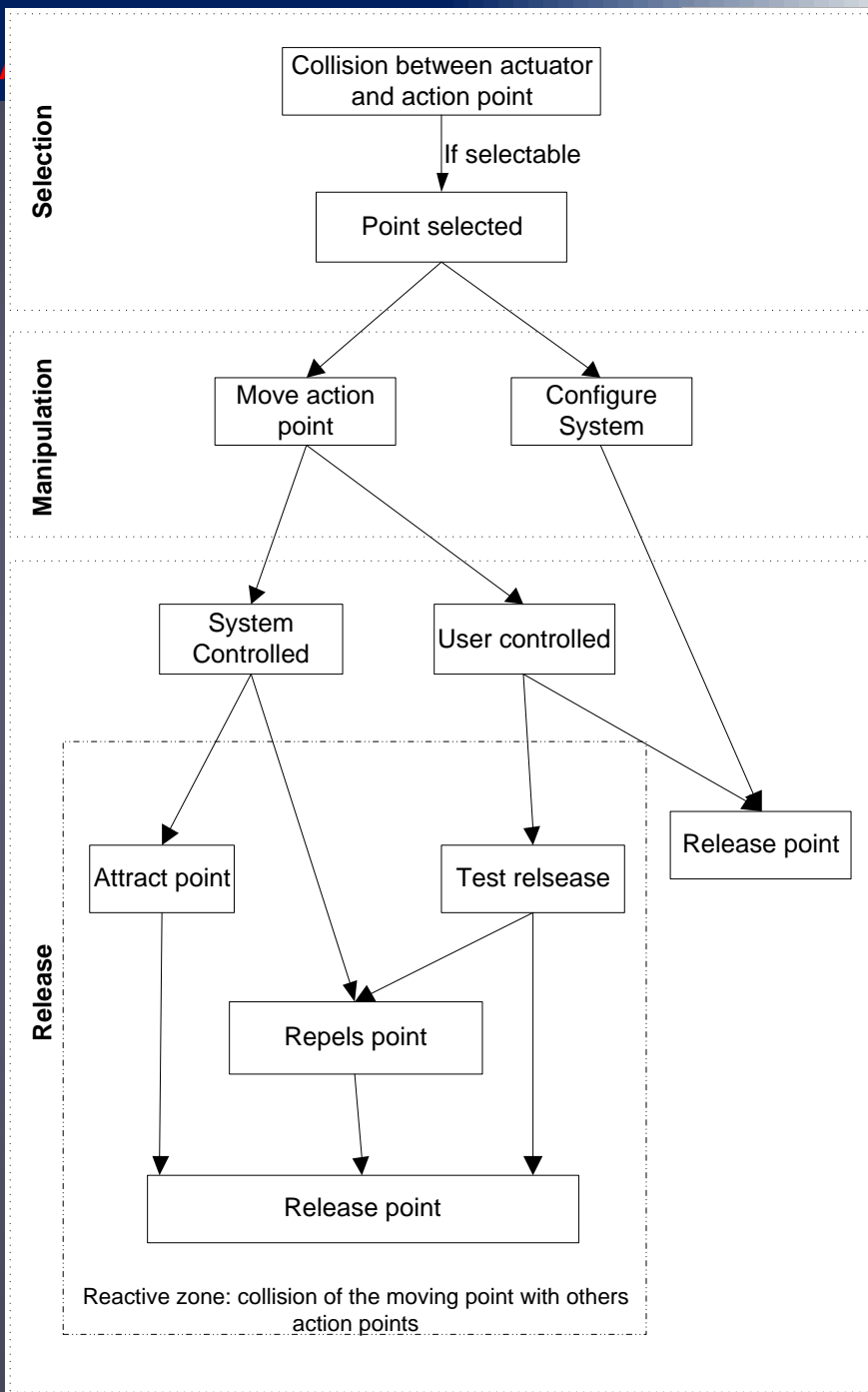
- Outros comandos vide manual do basAR!

# O que é click?

- Seleção → Ação → Liberação
- **Selecionar** um ponto onde a **ação** é alterar o fluxo de execução para outro estado, onde as ações e atributos dos pontos podem ser alterados.

# Mapa de atitudes:

- Clicando um ponto
  - Configuração do sistema
    - Modelos do ponto
    - Posicionamento do ponto
      - Animações simuladas
    - Matrizes de posicionamento



# Mudança de estado

- A mudança de estado pode ser realizada pelo sucesso dos comandos de movimentação (Vide tutorial sobre movimentação) ou utilizando um ponto com uma **ação de mudança de estado**.
- Com o **click** do atuador no ponto de mudança de estado é possível **mudar os atributos e reconfigurar a atitude e exibição** do ambiente.

# Tipos de estados

## Estado de trabalho

- Este tipo de estado espera uma ação ser completada para que ocorra mudança de estado

```
BEGIN_STATE 4  
    1 ...  
    2 ...  
END_STATE
```

## Estado de configuração

- Este tipo de estado, após aplicar as alterações de comportamento, altera automaticamente o estado.

```
BEGIN_STATE 4  
    1 ...  
    2 ...  
END_STATE GO_TO 5
```

# Comando que habilita mudança de estado

- Mudar estado (Change State)
  - CHGST

Point ID

CHGST

ShowMode

NextState

Audio

Overplay

- 1 CHGST ONLY\_OBJECT 4

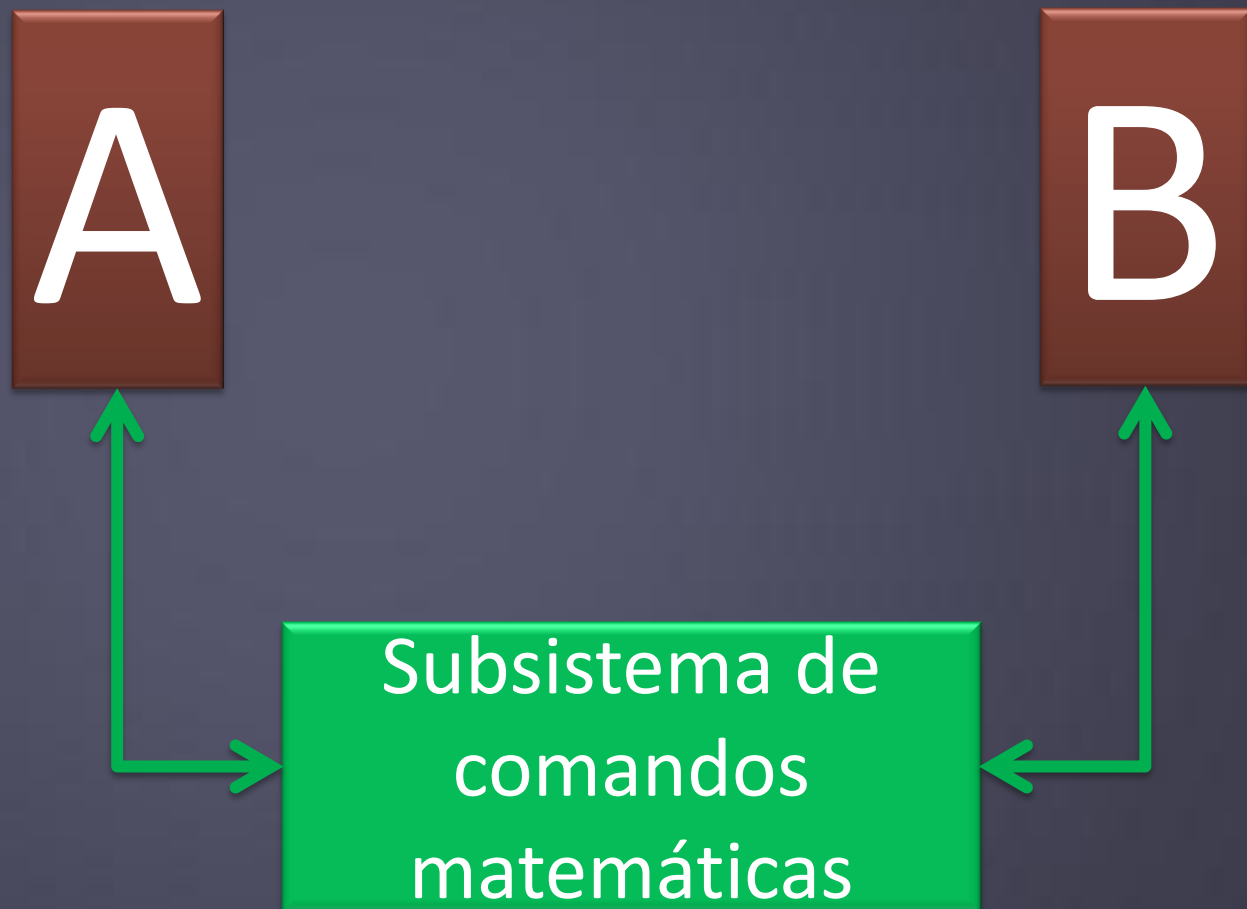
# Funções de controle matemáticas

- Permitem a construção de lógicas para auxiliar no controle do fluxo do programa.
- Permitem loops (For, while, do-while)
- Permitem colocar peso nos nós ( Inteligência Artificial)



# Variáveis por ponto

- Cada ponto tem uma variável A e B.



# Comandos entre as variáveis

- Carregar Valor:
- Adicionar valor
- Subtrair entre variaveis
- Swap
- Inverter sinal
- Multiplicar
- Comparar

# Comandos de carregar valor

- Carregar valor na variável A.  $A \leq \text{Value}$

– LOADA

Point ID

LOADA

Value

– 1 LOADA 25.5

- Carregar valor na variável B.  $B \leq \text{Value}$

– LOADB

Point ID

LOADB

Value

– 1 LOADB 30

# Comandos para adicionar valor

- Somar valor em A.  $A \leq A + \text{Value}$

– ADDA

Point ID

ADDA

Value

– 1 ADDA 15

- Somar valor em B.  $B \leq B + \text{Value}$

– ADDB

Point ID

ADDB

Value

– 1 ADDB 35

# Comandos para subtrair entre variáveis

- Subtrair B de A.  $A \leq A - B$

– AMB

Point ID

AMB

– 1 AMB

- Subtrair A de B  $B \leq B - A$

– BMA

Point ID

BMA

– 1 BMA

# Comandos para inverter sinal

- Inverte o sinal de A.  $A \leftarrow -A$

— NEGA

Point ID

NEGA

— 1 NEGA

- Inverte o sinal de B.  $B \leftarrow -B$

— NEGB

Point ID

NEGB

— 1 NEGB

# Comandos para multiplicar valor

- Multiplica valor em A.  $A \leq A * \text{Valor}$

- MULA

Point ID

MULA

Value

- 1 MULA 5

- Multiplica valor em B.  $B \leq B * \text{Valor}$

- MULB

Point ID

MULB

Value

- 1 MULB 3

# Comando de troca entre variáveis

- Trocar variáveis.

$A \leq B$  e  $B \leq A$

– SWAB

Point ID

SWAB

– 1 SWAB



# Comandos de comparação

- Compara A com B. Com condições. Se comparação é válida muda de estado.
  - $A > B$  ( GREATER)
  - $A < B$  ( LESSER)
  - $A = B$  (EQUALS)
  - CMP

Point ID

CMP

TYPE

NextState

- 1 CMP LESSER 7

# Comandos de comparação

- Compara A com Value. Com condições. Se comparação é válida muda de estado.

- $A > \text{Value}$  ( GREATER)

- $A < \text{Value}$  ( LESSER)

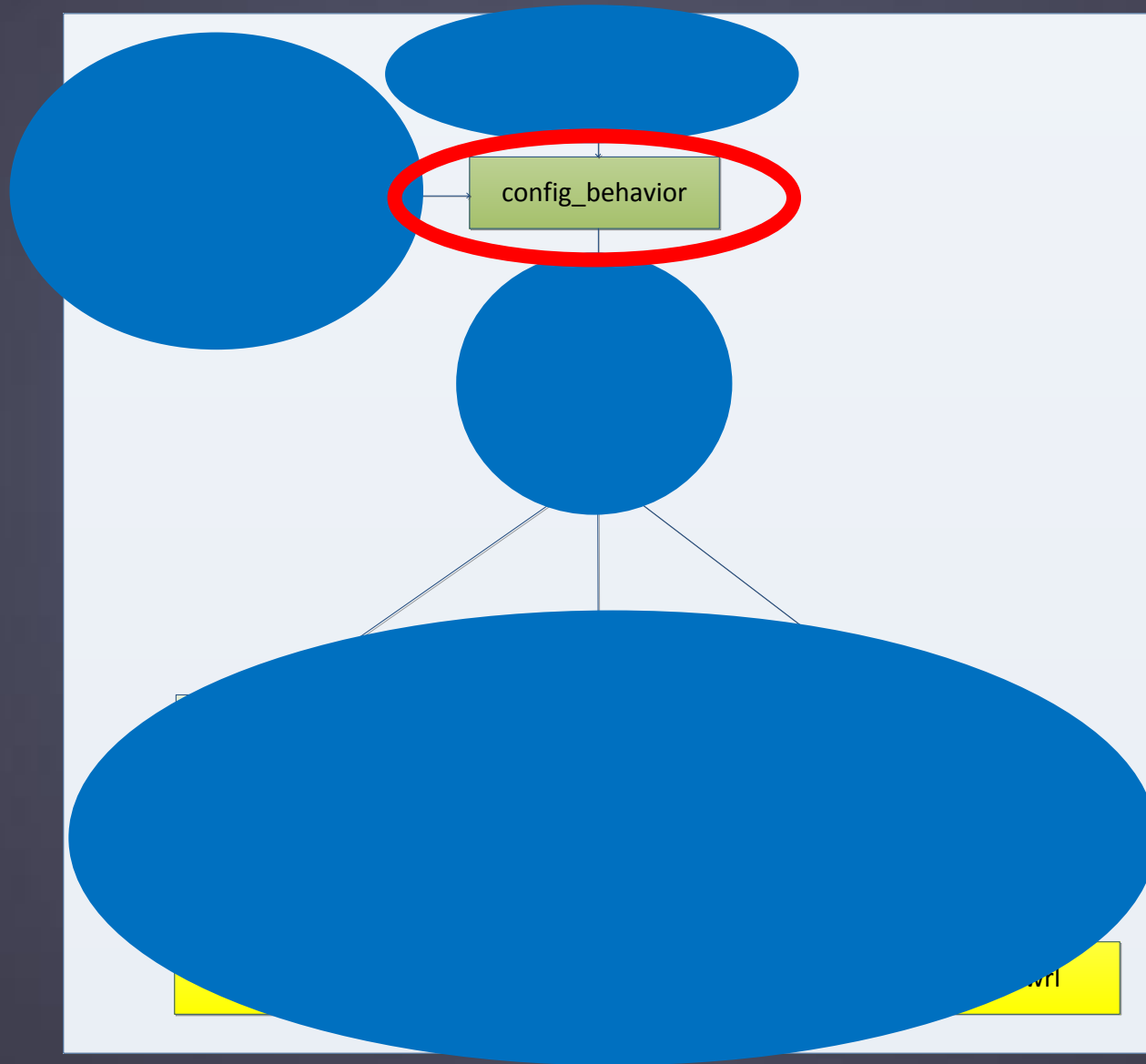
- $A = \text{Value}$  (EQUALS)

- CMP

Point ID	CMPV	Value	TYPE	NextState
----------	------	-------	------	-----------

- 1 CMP 35.5 LESSER 7

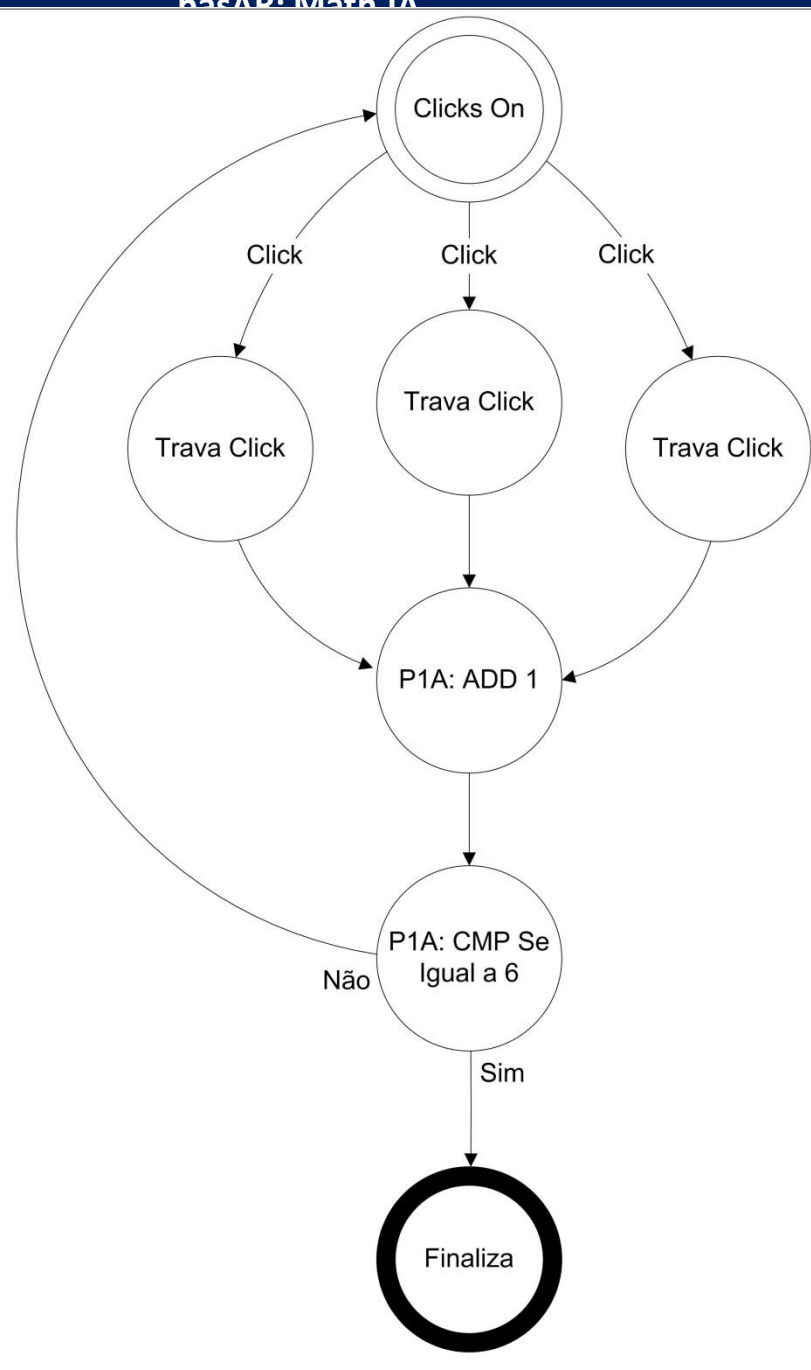
# Comportamento



# Exemplo de Loop

- Exemplo 1. Criar loop para contar Clicks.
- Arquivo de comportamento:  
**{basAR}/Data/config\_behavior**
- Clicando em qualquer ponto soma contador e quando houver um total de 6 clicks para a aplicação

# Fluxograma



# Arquivo de Comportamento

```
BEGIN_STATE 1
    1 CHGST ONLY_BALL 2
    2 CHGST ONLY_BALL 3
    3 CHGST ONLY_BALL 4
```

```
END_STATE
```

```
BEGIN_STATE 2
    1 STAT ONLY_BALL
END_STATE GO_TO 5 AFTER 1
```

```
BEGIN_STATE 3
    2 STAT ONLY_BALL
END_STATE GO_TO 5 AFTER 1
```

```
BEGIN_STATE 4
    3 STAT ONLY_BALL
END_STATE GO_TO 5 AFTER 1
```

```
BEGIN_STATE 5
    1 ADDA 1
END_STATE GO_TO 6
```

```
BEGIN_STATE 6
    1 CMPV 6 EQUALS 7
END_STATE GO_TO 1
```

```
BEGIN_STATE 7
    1 STAT ONLY_BALL
    1 STAT ONLY_BALL
    1 STAT ONLY_BALL
END_STATE
```

# RESUMO

# Resumo comandos utilizados

- **STAT**: [PointID] **STAT** [ShowMode] <AUDIO> <OVER?>
- **CHGST**: [PointID] **CHGST** [ShowMode] [NextState] <AUDIO> <OVER?>
- **LOADA**: [PointID] **LOADA** [Value]
- **LOADB**: [PointID] **LOADB** [Value]
- **ADDA**: [PointID] **ADDA** [Value]
- **ADDB**: [PointID] **ADDB** [Value]
- **AMB**: [PointID] **AMB**
- **BMA**: [PointID] **BMA**
- **NEGA**: [PointID] **NEGA**
- **NEGB**: [PointID] **NEGB**
- **MULA**: [PointID] **MULA** [Value]
- **MULB**: [PointID] **MULB** [Value]
- **CMP**: [PointID] **CMP** [GREATER/LESSER/EQUALS] [NEXTSTATE]
- **CMPV**: [PointID] **CMPV** [Value] [GREATER/LESSER/EQUALS] [NEXTSTATE]



# Ajuda online

- Videos exemplificando cada comportamento disponível e os arquivos utilizados neste tutorial estão disponíveis em:

<https://sites.google.com/site/christophercerqueira/projetos/ear/basar/cursos>



# Math.IA

## Curso basAR

Christopher Shneider Cerqueira – [christophercerqueira@gmail.com](mailto:christophercerqueira@gmail.com)

Claudio Kirner – [ckirner@gmail.com](mailto:ckirner@gmail.com)

<https://sites.google.com/site/christophercerqueira/projetos/ear/basar>

<http://www.ckirner.com/basar>