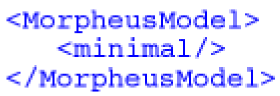
## Miscellaneous models

# Minimal model




Minimal valid XML model

### Introduction

This example does nothing - expect being the minimal valid Morpheus model.

Such a model is generated when choosing `File` → New.

### Model description

The basic model only includes the required nodes `MorpheusModel`, `Description`, `Space` and `Time`. Their required nodes and attributes are added recursively, such as `Lattice class` and `StopTime value`.

### Things to try

- Invalidate this minimal model by editing it (but keeping it well-formed). When opening this model in `morpheus-gui`, it triggers a warning saying that what went wrong, and how it was solved. Check the `Fixboard` to see the changes that are made to model.

### Model

h Minimal.xml |h

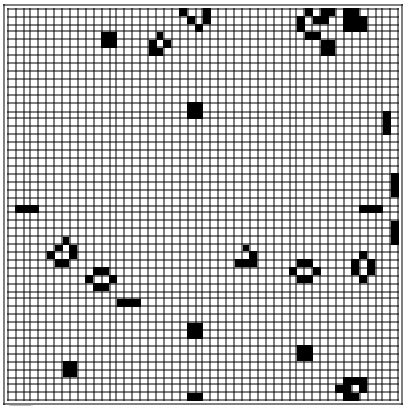
```xml
<?xml version='1.0' encoding='UTF-8'?>
<MorpheusModel version="1">
    <Description>
        <Title></Title>
    </Description>
    <Space>
        <Lattice class="linear">
            <Size value="0 0 0"/>
```

```
        </Lattice>
    </Space>
    <Time>
        <StartTime value="1.0"/>
        <StopTime value="1.0"/>
    </Time>
</MorpheusModel>
```

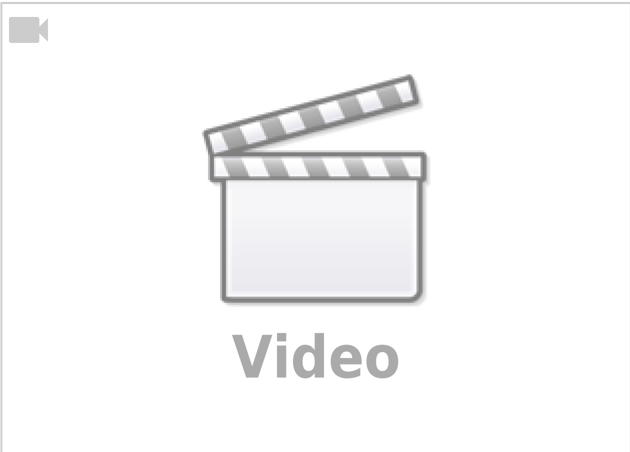
In Morpheus GUI: `File` → New.

---

## Game of Life: Cellular Automata



Conway's Game of Life




### Introduction

This example models probably the best-known classic cellular automaton (CA) model: Conway's Game of Life.

It shows an alternative use of `System` for synchronous updating of `Equations`.

### Model description

In this model, the lattice is filled with cells of size 1. Each cell counts the number of neighboring cells that are 'alive' and acts accordingly. The rules that make up the Game of Life are implemented in a `System` of `Equations` in which all `Equations` are updated synchronously.

#### Things to try

- Change the `Neighborhood` from a Moore (2nd order) to von Neumann (1st order).

#### Model

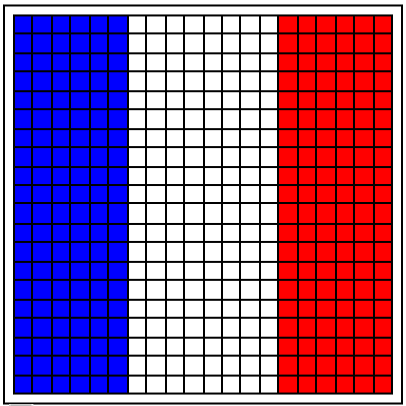
h GameOfLife.xml |h

```
extern>http://imc.zih.tu-dresden.de/morpheus/examples/Miscellaneous/GameOfLife.xml
```

In Morpheus GUI: `Examples → Miscellaneous → GameOfLife.xml`

---

## French Flag: Morphogen gradient



Wolpert's French Flag

#### Introduction

This example shows Wolpert's classical French Flag model. Depending on the local concentration of a morphogen, cells adopt one of three cell types based on internal thresholds.

#### Model description

The model sets up a morphogen gradient in the x direction PDE. Note that no diffusion is used, since we use the steady-state solution of diffusion.

The cells in `CellType` register the (average) local morphogen concentration using `PDEReporter`. Based on the specified threshold values, they choose an identity `I` as defined in the `Equation`.

Note that this model is not time-dependent. `Time` is therefore set from `StartTime` 0 to `StopTime` 0.

#### Things to try

- Change the physical length of the domain by editing `Space → NodeLength` that controls the physical size per lattice site.
- Change the model such that the morphogen gradient is set up by production and diffusion, using `Diffusion` and a `System` with `DiffEqn`. That is, change the model into a time-dependent model.

#### Model

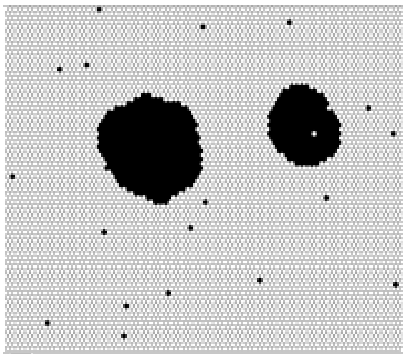
h FrenchFlag.xml |h

```
extern>http://imc.zih.tu-dresden.de/morpheus/examples/Miscellaneous/FrenchFlag.xml
```

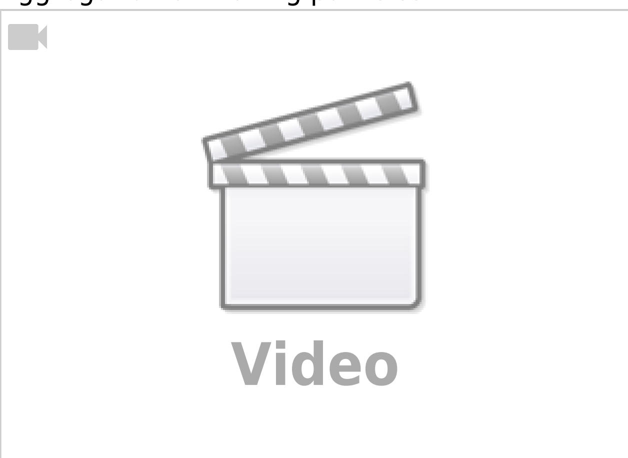
In Morpheus GUI: `Examples → Miscellaneous → FrenchFlag.xml`.

---

## Particle Aggregation: FlipCells



Aggregation of moving particles



#### Introduction

This models approximates an interacting particle system (IPS) model of particle aggregation. Each black dot represents a particle that moved due to spin flips with random neighbors. The particles perform random walks in which the probability of moving depends on the number of neighboring cells.

#### Model description

Each lattice site (white or black) counts the number of particles (black neighboring sites) using a `NeighborsReporter`.

The probability of movement of each particle is made dependent on its number of neighbors by using it in the `Condition` of `FlipCells`. When this condition is satisfied, the particle changes positions with a random neighboring lattice site.

A `PopulationReporter` is used to return the fraction of isolated black particles. This number is logged and plotted using the `Logger`.

#### Things to try

- Change the parameter p.

#### Model

h ParticleAggregation.xml |h

```
extern>http://imc.zih.tu-dresden.de/morpheus/examples/Miscellaneous/ParticleAggregation.xml
```

In Morpheus GUI: `Examples → Miscellaneous → ParticleAggregation.xml`

---