Divide and Conquer Implementation Pattern

Authors: Nigel T. Crowther

Date: 07-03-2023

Introduction

This project is an example of the **Divide and Conquer Pattern** applied to business rules. The pattern helps reduce complexity of rule logic.

# Divide and Conquer Pattern

## Pattern Definition

The Divide and Conquer Pattern splits one big decision table into smaller decisions which combine the overall result. See below.



Figure : Divide and Conquer

## Pattern Advantages

This pattern helps split one long decision table with many columns into smaller constituent tables. It helps reduce duplicate conditions, encourages condition reuse, and simplifies business logic.

## Pattern Disadvantages

Excessive use of this pattern creates a proliferation of intermediate tables that may affect performance and maintenance.

## Divide and Conquer Pattern – Simple Implementation

Consider a rule application that categorises planets for habitability. The rule inputs are:

* **Oxygen** - As a percentage of total atmosphere
* **Surface Pressure** – In millibars
* **Temperature** - Average daily temperature in degrees centigrade

The output from the rules is an indication of the habitability of the planet.

Our initial design might be developed as a decision table shown below:

|  |  |  |  |
| --- | --- | --- | --- |
| **Oxygen** | **Surface Pressure** | **Temperature** | **Habitability** |
| At least 15% | More than 1000mb | More than 10 | **Good** |
| 5 to 15% | 100-999 | More than 0 | **Bearable** |
| otherwise |  |  | Deadly |

Figure : Planet Habitability

The decision table in Figure 2 is not complete and once all values are added it will be a large and difficult table to understand. To simplify it, we could decompose the table into smaller tables which when combined create the overall result. The design using DMN is as follows:

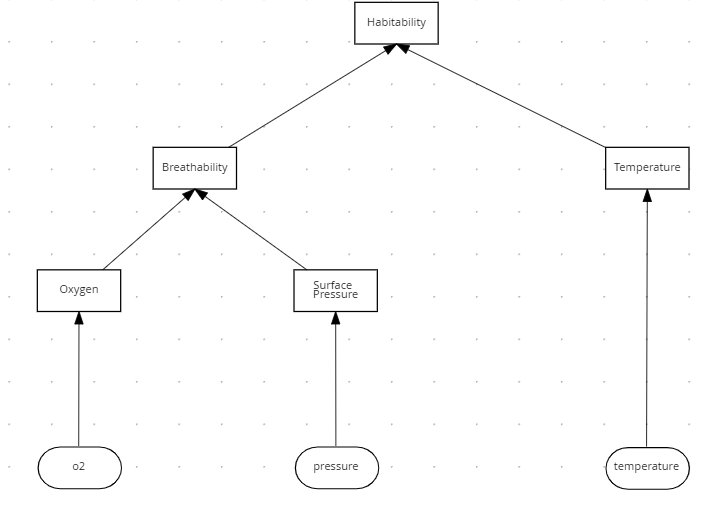


Figure : Habitability Tables

We decompose each column in Figure 2 into separate tables:

### Surface Pressure Table

|  |  |
| --- | --- |
| **Mb** | **Pressure** |
| >=1000 | Optimal |
| 100-999 | Bearable |
| < 100 | Deadly |

Figure : Surface Pressure Table

### Oxygen Table

|  |  |
| --- | --- |
| **Mb** | **Pressure** |
| >=15 % | Optimal |
| 5 - 15% | Bearable |
| < 5% | Deadly |

Figure : Oxygen Table

### Breathability Table

|  |  |  |
| --- | --- | --- |
| **Oxygen** | **Pressure** | **Breathability** |
| Optimal | Optimal | Optimal |
| Bearable | Bearable | Bearable |
| Deadly | Deadly | Deadly |

Figure : Breathability Table

### Temperature Table

|  |  |
| --- | --- |
| **Temperature** | **Classification** |
| > 50 | Deadly |
| > 0 | Bearable |
| >10 | Temperate |

Figure : Temperature Table

### Habitability

We now join the Breathability and Climate tables which produces the overall habitability decision:

|  |  |  |
| --- | --- | --- |
| **Breathability** | **Climate** | **Habitability** |
| Good | Temperate | Good |
| Poor | Cold | Poor |
| Poor | Hot | Poor |
| Otherwise |  | Fair |

Figure : Habitability

The **habitability** table above has the same behaviour as Figure 2 except that it is easier to understand and maintain. An additional advantage is that the smaller tables can be reused in other decisions.

## Divide and Conquer Pattern – Advanced Implementation

The DMN diagram we created in figure 1 is simplified. If we were to add additional factors for planetary habitability such as radiation, the DMN would rapidly become too complex for a single diagram. See below:

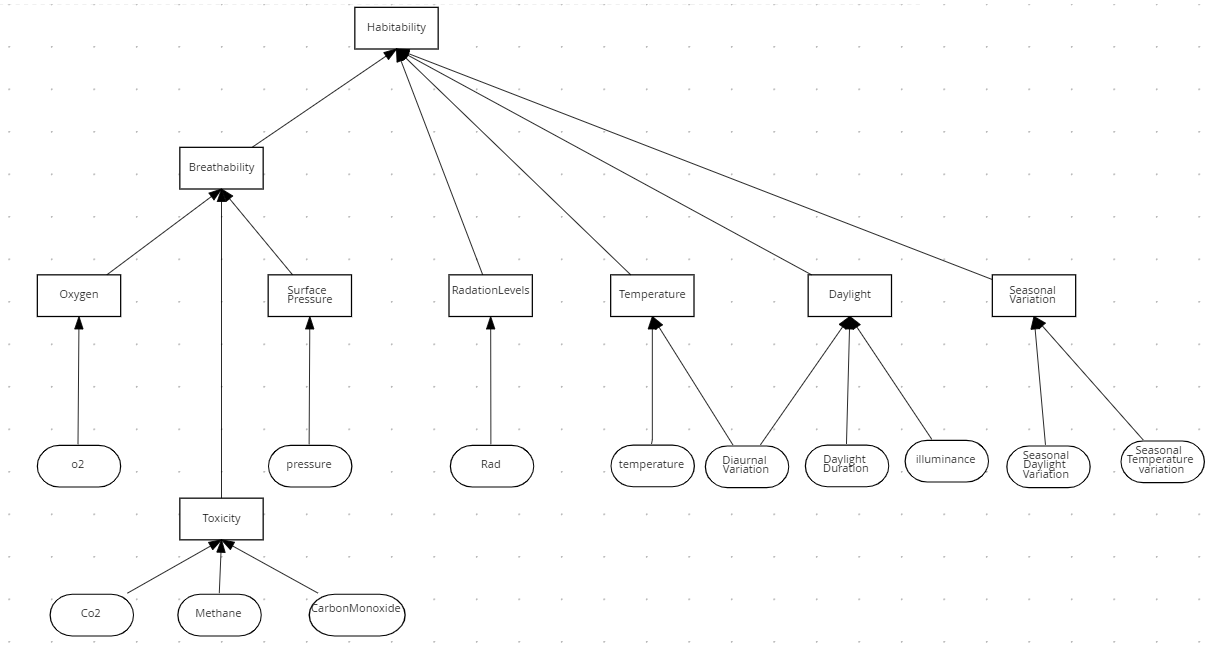


Figure 9: Habitability with additional parameters

In such cases as above, it would be helpful to decompose the diagram into smaller diagrams. We can accomplish this using Decision services.