

# Knowledge Graphs

## Lecture 5 – Ontological Engineering for Smarter Knowledge Graphs

### 5.2 How to design your own Ontology

**Prof. Dr. Harald Sack & Mary Ann Tan**

FIZ Karlsruhe – Leibniz Institute for Information Infrastructure

AIFB – Karlsruhe Institute of Technology

**Autumn 2023**



# Knowledge Graphs

## Lecture 5: Ontological Engineering for Smarter Knowledge Graphs

### 5.1 Beyond the Limits of OWL

Excursion 7: The Semantic Web Rule Language SWRL

### 5.2 How to design your own Ontology

### 5.3 How to design better Ontologies

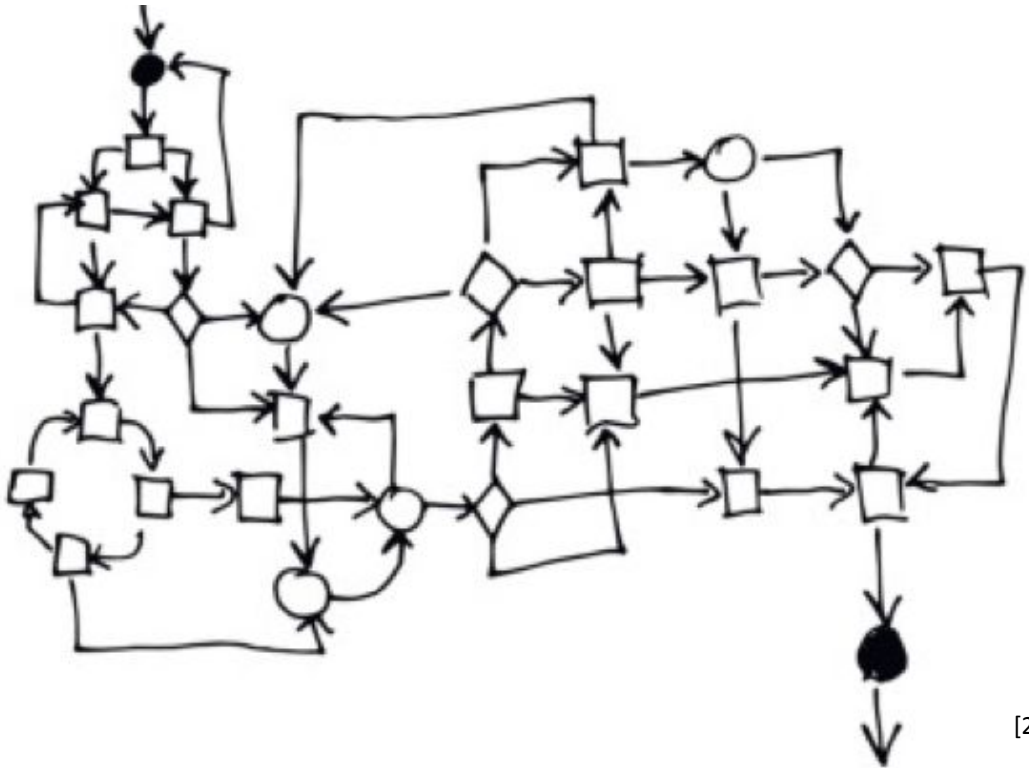
### 5.4 Ontological Engineering

### 5.5 Knowledge Graph Construction

### 5.6 Ontologies & Knowledge Graphs – Best Practices

# The Ontology Development Process

**SOMETHING**

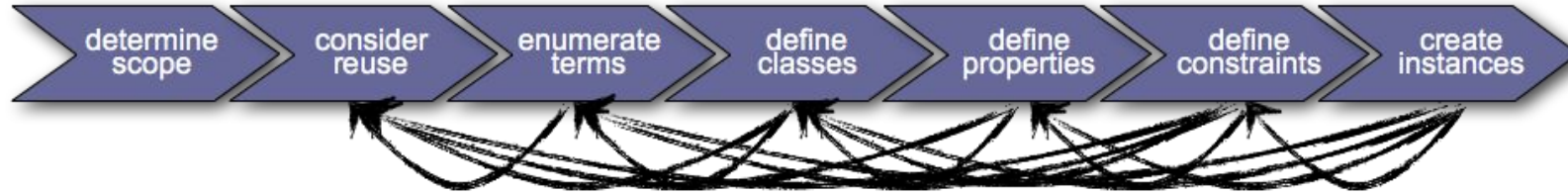


[2]

**Great Ontology**

# The Ontology Development Process

(Noy, McGuinness, 2000)



- In practice, the ontology development process is an **iterative Process** that **repeats continuously** and improves the ontology
- There are always **different approaches** for modelling an ontology
- The designated application decides about the modelling approach

“There is no one correct way to model a domain.  
There are always viable alternatives.”

## Determine Domain and Focus



- Which **Domain** should be covered by the ontology?
- **What** should the ontology be used for?
- **Who** will use and maintain the ontology?
- What types of **Questions** should be answered by the knowledge represented in the ontology?
- Formulation of **Competency Questions**!

## Determine Domain and Focus



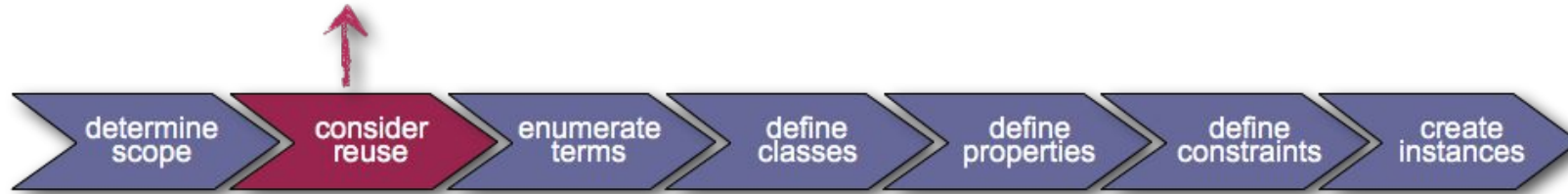
### **Competency Questions** (Example: Wine Ontology)

- Which properties of the wine should be considered for modelling?
- Is Bordeaux a white wine or a red wine?
- Does a Sauvignon Blanc match with fish?
- Which wine matches best with grilled vegetables?
- Which properties of a wine do influence whether it matches with a specific dish?
- Does the bouquet of a wine change with time?
- Does the bouquet of a wine change with storage conditions?
- ...

These Questions might change  
within the ontology life cycle.



# Consider Reuse



Why should we consider reuse?

- In order to save **cost**
- In order to apply **tools** that are applied to other existing ontologies also to our own ontology
- In order to reuse ontologies that have been validated by their application

If you don't find a suitable ontology or if the adaption is too complex then create a new ontology!

# Develop a Terminology



- Which **concepts** do we need to represent?
- Which **properties** do these concepts have?
- **What** do we want **to say** about these concepts?

## Example: **Wine Ontology**

- *wine, grape, winery, location,...*
- *a wine's colour, body, flavor, sugar content,...*
- *subtypes of wine: white wine, red wine, Bordeaux wine,...*
- *types of food: seafood, meat, vegetables, cheese, dessert,...*
- ...



# Define Classes and Class Hierarchies

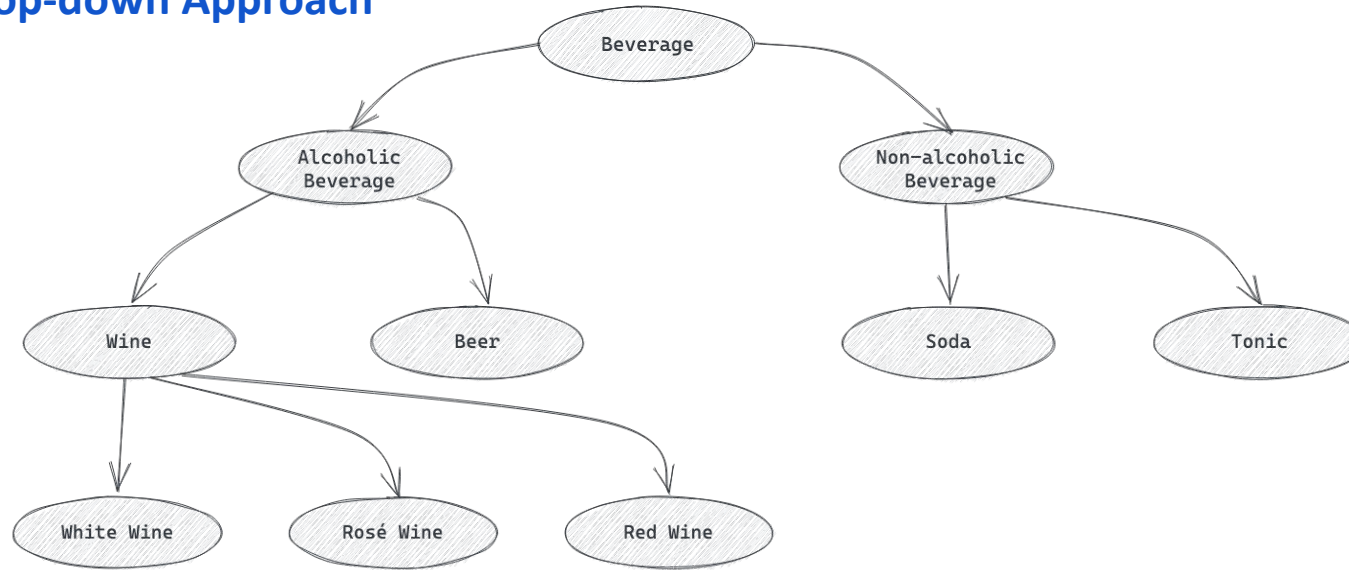


- **Classes** are concepts in the designated domain
  - *class of wines*
  - *class of wineries*
  - *class of red wines*
  - ...
- Classes are collections of objects with **similar properties**.
- Choose a **top-down / bottom-up / middle-out** approach to model class hierarchies.

# Define Classes and Class Hierarchies



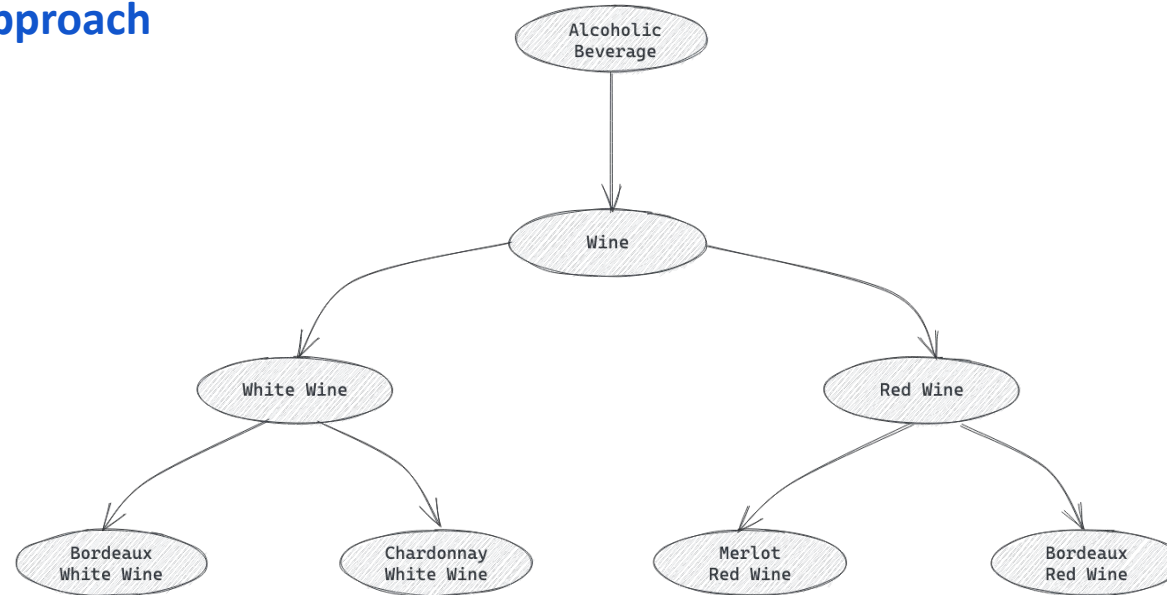
## Top-down Approach



# Define Classes and Class Hierarchies



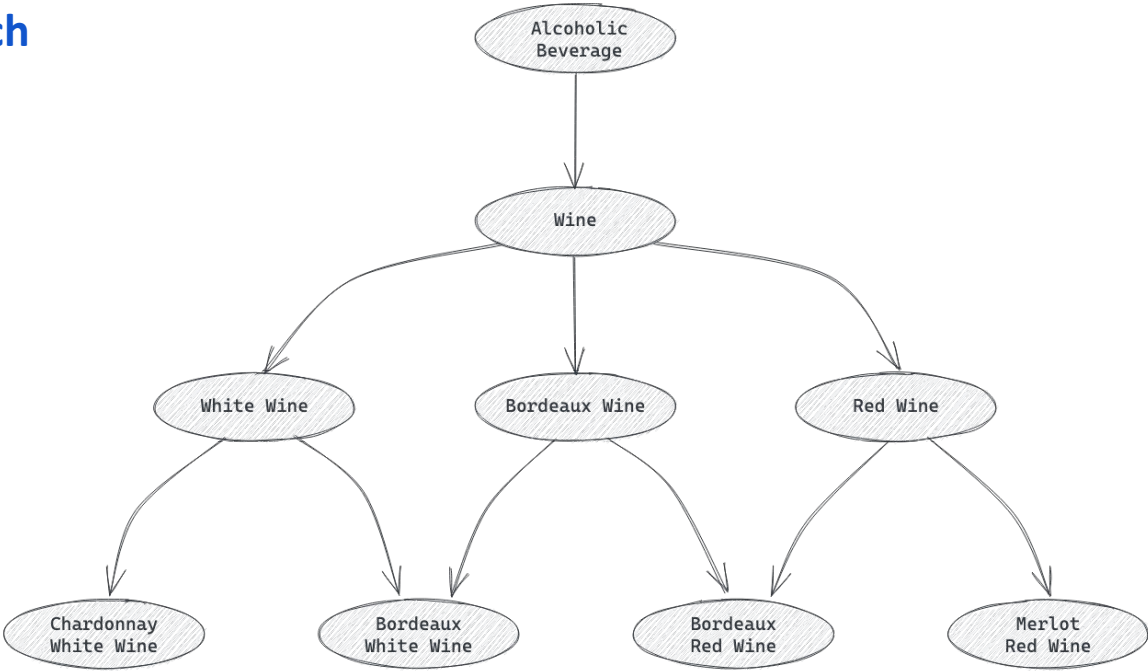
## Bottom-up Approach



# Define Classes and Class Hierarchies



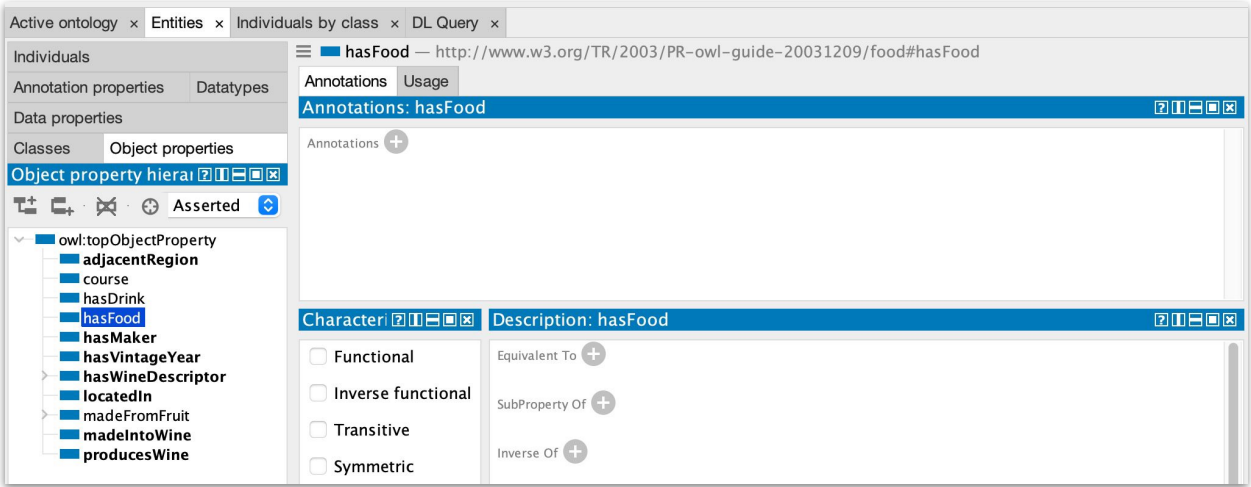
## Middle-out Approach



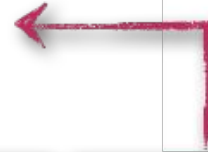
# Define Properties



**Properties** in a class definition describe attributes of instances  
*every wine has a colour, residual sugar, producer, etc...*

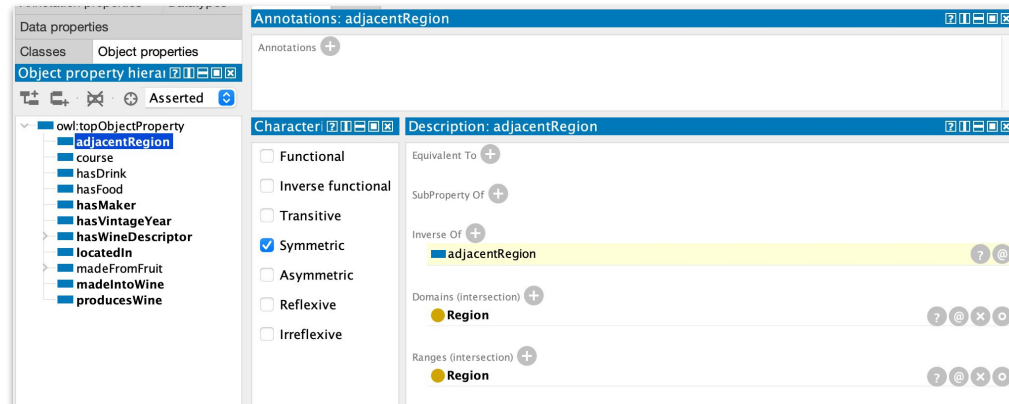


# Define Property Constraints



**Property constraints** (restrictions) describe or restrict the set of possible property values

- *The name of a wine is a String*
- *The producer is an instance of Winemaker...*





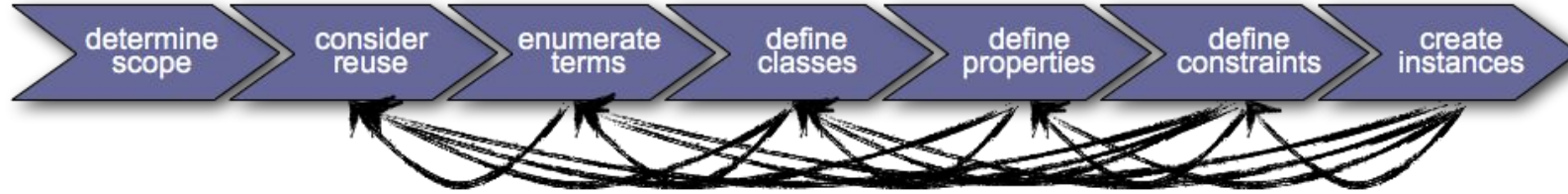
# Define Class & Property Instances



- Create **instances for the classes**:
  - Every **class** directly becomes the **type** of its instances.
  - Every **superclass** of a direct type is also type of its instances.
- Create **instances for properties**, i.e. the assignment of property values for the instances according to the given constraints.
- *“The glass of red wine that I drank last supper...”*

# The Ontology Development Process

(Noy, McGuinness, 2000)



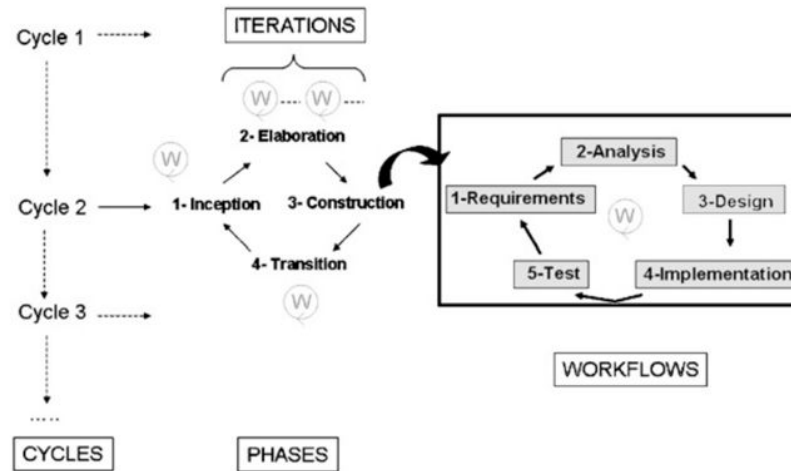
Ontology development in practice is an **iterative process** that **repeats continuously** and improves the ontology.

“There is no one correct way to model a domain.  
There are always viable alternatives.”

# Unified Process for Ontology Building

De Nicola, Missikoff, Navigli (2005)

- Development is divided into **Cycles**, which are subdivided into four **Phases** of **Iterations** (*Inception, Elaboration, Construction, Transition*). Each iteration results in a new prototype.



- Each iteration consists of five workflows (*Requirements, Analysis, Design, Implementation, Test*).

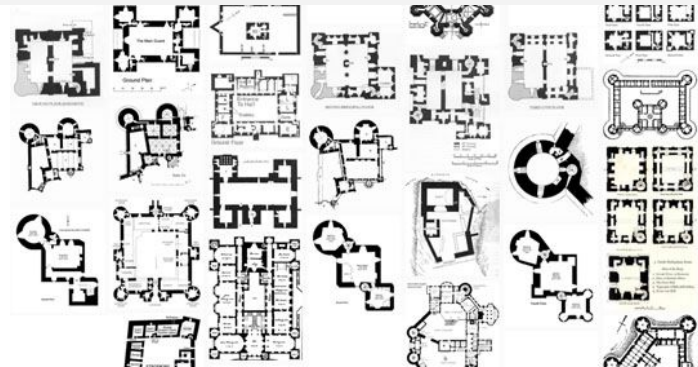
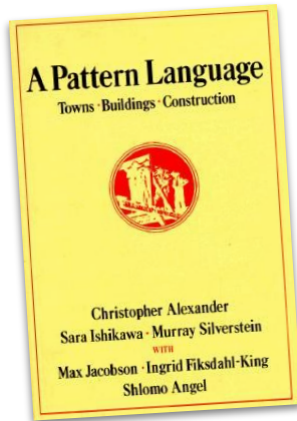
# Ontology Design Patterns

Gangemi (2005)

Adapting a design idea originally from architecture

- Recurring modelling problems
- Providing a set of adaptable standard solutions
- *A “pattern” is a solution to a problem in a given context (Christopher Alexander)*

The elements of this language are entities called patterns. Each pattern describes a problem that occurs over and over again in our environment, and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice. — *Christopher Alexander*



# How to design better Ontologies

Next Lecture...



### Bibliographic References:

- Natasha Noym Deborah McGuinness (2001). [\*Ontology Development 101: A Guide to Creating Your First Ontology.\*](#)
- De Nicola, A., Missikoff, M., Navigli, R. (2005). [\*A Proposal for a Unified Process for Ontology Building: UPON.\*](#) In: Andersen, K.V., Debenham, J., Wagner, R. (eds) Database and Expert Systems Applications. DEXA 2005. Lecture Notes in Computer Science, vol 3588. Springer, Berlin, Heidelberg.
- Gangemi, A., & Presutti, V. (2009). [\*Ontology Design Patterns. Handbook on Ontologies,\*](#) pp 221–243.
- Christopher Alexander, Sara Ishikawa, Murray Silverstein, Max Jacobson, Ingrid Fiksdahl-King, Shlomo Angel, [\*A Pattern Language - Towns, Buildings, Construction,\*](#) Oxford University Press, 1977.

### Picture References:

- [1] “A Scifi movie poster of "Planet Mars - the isle of the Dead". A small rover crosses the lonely Martian dessert towards the isle of the dead, on board we see two silent astronauts. Some zombies are chasing after the astronauts.”, created via ArtBot, Deliberate, 2023, [CC-BY-4.0], <https://tinybots.net/artbot>
- [2] The Software Development Process, Geek & Poke, <http://geekandpoke.typepad.com/geekandpoke/2012/01/simply-explained-dp.html>
- [3] “On this scifi movie poster we see the vibrant construction site of a gigantic space ship in the vast deserts of planet Mars exposing many small details.”, created via ArtBot, Deliberate, 2023, [CC-BY-4.0], <https://tinybots.net/artbot>