Semantic Web is an extension of the traditional web where data is structured in such a way that it can be stored, understood, shared and reused across different applications, platforms and communities. Its primary goal is to transform the web from a collection of linked documents into a web of linked data.
It uses standards such as RDF (Resource Description Framework), OWL (Web Ontology Language) and SPARQL to and query data.

Ontologies in the context of the Semantic Web are formal models that define the relationships between different concepts and entities within a specific domain. An ontology provides a shared vocabulary and a set of roles (axioms) that describe how concepts are related to one another, allowing machines to understand and process data consistently. It contains classes, properties, instances and axioms.

RDF is a standard for representing structured information about resources on the web. It uses a directed graph model to express relationships between entities in the form of subject-predicate-object triples.

Knowledge Base RDFs can be combined to form a KB, which is a collection of RDF triples or graphs that represent a dataset. Each KB is assigned a name (URI) and 2 KBs should never have the same name.

Cross-Referencing is when a KB can make statements about entities defined in other KBs.

Namespace is a way to organize and disambiguate URIs within a RDF. It defines a common prefix for a group of URIs to prevent naming conflicts between different vocabularies or datasets.

Qname A qualified name is a shorthand notation for referring to URIs by using a common prefix and a local name.

Blank Node A blank node is a concept in RDF, representing an entity in the RDF data model that does not have a URI. Instead, blank nodes are used locally in a RDF graph for internal references without being explicitly identified outside the graph.

TURTLE Terse RDF triple language is a textual syntax for expressing RDF data. It is a compact and human-readable way to write RDF triples, which consist of subject, predicate and object.

Base URI is a reference URI for URIs within a document of dataset for simplification and consistency.

Multivalued Relationship refers to a case where an entity (subject) has multiple values for the same property (predicate). For example, in the context of a recipe, the ex:Pasta has multiple ex:hasSauce such as pesto, carbonara, tomato and etc.

```
#Define a namespace prefix "ex"
@prefix ex: <http://www.recipes.es/european/italy>.
#A triple that uses the prefix and a blank node
ex:Pasta ex:hasSauce _:pesto, _:carbonara .
#Blank node description properties
_:pesto ex:sauceName "Pesto" ;
        ex:hasIngredient _:pesto1 .
_:pesto1 ex:name "Basil" ;
         ex:amount "80g" .
```

RDF Schema RDFS vocabulary, which is a extension of RDF, provides a basic set of constructs to describe relationships and properties of resources in a semantic web environment. Some basic terms include: rdfs:Class, rdfs:subClassOf, rdfs:domain, rdfs:range and rdfs:label.

Ontology (Formal definition) $O = (TBox, ABox)$

- **The ABox** contains facts about specific instances of the concepts and the roles defined in the TBox. It is the "data" part of the ontology. It contains instances and assertions which are statements that describe how individuals relate to concepts and roles.
- **The TBox** is responsible for defining the vocabulary of the ontology, i.e., the structure of the domain knowledge. It contains concepts (classes) (abstract groups or sets that define categories of things in the domain), also called atomic concepts, usually grouped into a set N_C. Also contains roles (properties) which are relationships between concepts, they describe how atomic concepts are related to each other and are grouped into a set N_R. Finally, it contains axioms which are logical statements or constraints that describe the structure of concepts and their relationships.

ALC is a basic form of description logic that allows to create complex concepts by combining atomic concepts using logical operators and quantifiers.
- Atomic Conc: If $A \in N_C$, then $A \in ALC$.
- Roles: If C, D are ALC concepts and $r \in N_R$, then the following are ALC concepts:
  1) Logical Operators → conjunction $\sqcap$, negation, disjunction $\sqcup$
  2) Symbols: $\top$ (truth), $\bot$ (false)
  3) Quantifiers:
     - Existential Restriction ($\exists r.C$): This means "there exists a role r pointing to a concept C.
     - Value Restriction ($\forall r.C$): This means "for all roles r, they point to concept C.

Interpretation An interpretation $I = (\Delta^I, \cdot^I)$ consists of:
A) A non-empty domain Δ^I: This is the set of all possibles individuals that the interpretation considers. These individuals are the entities over which concepts and roles are defined.
B) An extension mapping $\cdot^I$: This mapping defines how atomic concepts, roles and individuals are interpreted within the domain. For every concept $A \in N_C$, $A^I \subseteq \Delta^I$. For every role $r \in N_R$, $r^I \subseteq \Delta^I \times \Delta^I$. For each individual a, $a^I \in \Delta^I$.

$\neg(\forall x.\phi) = \exists x.\neg\phi$; $\neg(\exists x\phi) = \forall x.\neg\phi$

General Concept Inclusion GCI is a formal expression of the form $C \sqsubseteq D$, where C and D are concepts. The expression asserts that every instance of concept C is also an instance of concept D (i.e., $C^I \subseteq D^I$ in any interpretation I).

The extension mapping follows some rules:
- Conjunction: $(C \sqcap D)^I = C^I \cap D^I$
- Disjunction: $(C \sqcup D)^I = C^I \cup D^I$
- Negation: $(\neg C)^I = \Delta^I \setminus C^I$
- $(\exists r.C)^I = \{d \in \Delta^I \mid$ there exists $e \in \Delta^I$ such that $(d, e) \in r^I$ and $e \in C^I\}$
- $(\forall r.C)^I = \{d \in \Delta^I \mid$ for all $e \in \Delta^I$, if $(d, e) \in r^I$, then $e \in C^I\}$

Concept Assertion is of the form $C(a)$, where C is a concept and a is an individual. A **role assertion** is of the form $r(a,b)$, where r is a role between individuals a and b.

PROPERTIES:
- An interpretation I is a model of the ontology O if it satisfies the TBox and ABox.
- Inconsistency: An ontology O is inconsistent if O has no model.
- Incoherency: An ontology O is incoherent if O has an unsatisfiable concept.
  ↳ A concept C is unsatisfiable if $C^I = \emptyset$ with respect to the ontology, for instance $C = \neg A \sqcap A$.

A) Subsumption: Concept C is subsumed by concept D with respect to a TBox T if, in all models I of T, $C^I \subseteq D^I$.

B) Equivalence: Concepts C and D are equivalent with respect to a TBox T if, in all models I of T, $C^I = D^I$.

C) Entailment: An ontology O entails a GCI $C \sqsubseteq D$, written $O \models C \sqsubseteq D$, if in all models I of T, $C^I \subseteq D^I$. Entailment also apply to concept and role assertions.

$T_1 = \{A = A_1 \sqcap A_2\}$
$T_2 = \{A = A_1, A = A_2\}$
TBox

¿Is concept A in T_1 equivalent to concept A in T_2?

PROOF To prove or check equivalence, we need to check whether both TBoxes have the same models. In other words, for every interpretation I, the definitions in T_1 must imply those in T_2 and viceversa.

$\boxed{T_1 \Rightarrow T_2}$ In T_1, we have $A = A_1 \sqcap A_2$, meaning $A^I = (A_1 \sqcap A_2)^I = A_1^I \cap A_2^I$. This means every instance of A is both in A_1 and A_2. However it does not imply that $A_1^I = A_2^I$ for every interpretation I, since A_1 and A_2 could have additional elements outside A. Thus, $A^I = A_1^I \cap A_2^I$ ($\neq A = A_1^I = A_2^I$), therefore T_1 does not imply T_2.

$\boxed{T_2 \Rightarrow T_1}$ In T_2, we have that $A = A_1$ and $A = A_2$, implying that $A^I = A_1^I = A_2^I$. This clearly implies that $A = A_1 \sqcap A_2$ because $A^I = A_1^I = A_2^I$ and hence $A^I = A_1^I \cap A_2^I$. So T_2 does imply T_1.

Conclusion: T_2 is stronger than T_1 because it forces A_1 and A_2 to be the same set, whereas T_1 only forces A to be the intersection of A_1 and A_2 without requiring A_1 and A_2 to be equal. Therefore, T_1 and T_2 are NOT EQUI

$T_1 = \{A \sqsubseteq A_1 \sqcap A_2\}$ $\not\models$ A in T_1 equiv to $T_2 = \{A \sqsubseteq A_1, A \sqsubseteq A_2\}$ A in T_2! $\boxed{\text{PROOF}}$ same intro as before. $\boxed{T_1 \Rightarrow T_2}$ In T_1, we have $A \sqsubseteq A_1 \sqcap A_2$ meaning $A^I \subseteq (A_1 \sqcap A_2)^I \subseteq A_1^I \cap A_2^I \Rightarrow A^I \subseteq A_1^I$ and $A^I \subseteq A_2^I$. Thus, T_1 implies T_2. $\boxed{T_2 \Rightarrow T_1}$ In T_2 we have that $A \sqsubseteq A_1$ and $A \sqsubseteq A_2$ implying that $A^I \subseteq A_1^I$, $A^I \subseteq A_2^I$. This clearly implies that $A^I \subseteq A_1^I \cap A_2^I$ and therefore $A \sqsubseteq A_1 \sqcap A_2$. Therefore T_1 and T_2 are equivalent.

$O = (T, A)$ with $T = \{A \sqsubseteq B, B \sqsubseteq C\}$ entails $A \sqsubseteq C$. $\boxed{\text{PROOF}}$ We need to show that in every inter I, the inclusion $A^I \subseteq C^I$ holds. We know that:
- $A \sqsubseteq B$ in $T \Rightarrow A^I \subseteq B^I \Rightarrow$ Every instance of A is also an instance of B, then $a \in A^I \Rightarrow a \in B^I$.
- $B \sqsubseteq C$ in $T \Rightarrow B^I \subseteq C^I \Rightarrow$ Every instance of B is also an instance of C, then $a \in B^I \Rightarrow a \in C^I$.

$\Rightarrow A^I \subseteq C^I$.

Lemma 1: If C is unsatisfiable w.r.t O, and $O \subseteq O'$ then C is unsatisfiable w.r.t O'. $\boxed{\text{PROOF}}$ We know that C is unsatisfiable w.r.t O which means that for every interpretation I, if I is an interpretation of O, then $C^I = \emptyset$. Besides we also know that $O \subseteq O'$, meaning O' contains all axioms of O and possibly more. Let's prove it by contradiction. Suppose C is satisfiable w.r.t O'. That means, exists an inter I' that satisfies both O' and C, meaning $C^{I'} \neq \emptyset$. Since $O \subseteq O'$, every inter I' of O' is also an inter I' of O because O' contains all axioms of O. By assumption, C is unsatisfiable w.r.t O, i.e for any model I of O, $C^I = \emptyset$, but since I' is a model of O', and therefore a model of O we must have $C^{I'} = \emptyset$ which contradicts the assumption that $C^{I'} \neq \emptyset$.

Lemma 2: If C is satisfiable w.r.t O, and $O' \subseteq O$ then C is satisfiable w.r.t O'. $\boxed{\text{PROOF}}$ We know that C is satisfiable w.r.t O, i.e exists an inter I s.t I satisfies C, i.e. $C^I \neq \emptyset$. Besides we also know that $O' \subseteq O$, which means O' contains fewer or equal axioms compared to O. We must show that there is some inter I' that satisfies both O' and C, i.e. $C^{I'} \neq \emptyset$. Since $O' \subseteq O$, the onto O' has fewer axioms than O. An inter that satisfies the more restrictive onto O should also satisfy the less restrictive onto O'. Since I is a model of O and satisfies C, i.e. $C^I \neq \emptyset$ and because $O' \subseteq O$, I must also be a model of O'. This is because O' contains a subset of the axioms in O, so satisfying axioms of O automatically satisfies the fewer axioms of O'. Therefore, the inter I that satis C w.r.t O also satis C w.r.t O', i.e $C^I \neq \emptyset$.

$\forall r.A \sqcap \forall r.\neg A$ is unsatisfiable, i.e. $(\forall r.A \sqcap \forall r.\neg A)^I = \emptyset$ for any model I of O. $\boxed{\text{PROOF}}$ We will prove it by contradiction, let's assume that $(\forall r.A \sqcap \forall r.\neg A)$ is satisfiable, i.e. exists an inter I and some individual x in Δ^I such that: $x \in (\ldots)^I = (\forall r.A)^I \cap (\forall r.\neg A)^I \Rightarrow x \in (\forall r.A)^I$ and $x \in (\forall r.\neg A)^I$. By def of $(\forall r.A)^I$: $\{d \in \Delta^I \mid \forall e \in \Delta^I, \text{if } (d,e) \in r^I, \text{then } e \in A^I\}$ It means all indi related to d via role r must belong to A^I. Similarly, $(\forall r.\neg A) = \{d \in \Delta^I \mid \forall e \in \Delta^I, \text{if } (d,e) \in r^I, \text{then } e \in (\neg A)^I \Leftrightarrow e \notin A^I\}$. This clearly proves a contradiction and therefore $(\forall r.A \sqcap \forall r.\neg A)$ is not satisfiable.

$\forall r.A \sqcap \exists r.\neg A$ is unsatis, i.e. $(\ldots)^I = \emptyset$ for any model O. $\boxed{\text{PROOF}}$ We proceed by proof of contra, by assuming that $(\ldots)$ is satisfi, i.e. exists an inter I and some indi x in Δ^I s.t: $x \in (\ldots)^I = (\forall r.A)^I \cap (\exists r.\neg A)^I$. By def *, all indi related to d via role r must belong to A^I, and $(\exists r.\neg A)^I$: $\{d \in \Delta^I \mid \exists e \in \Delta^I \text{ s.t } (d,e) \in r^I \text{ and } e \notin A^I\}$. This proves a contra, because from $x \in (\forall r.A)^I$ all objects related to d via r must be in A^I, i.e $e \in A^I$ and from $x \in (\exists r.\neg A)^I$, there must exist some e s.t $e \notin A^I$. Hence, $(\ldots)^I = \emptyset$ for any inter I.

$\exists r.(A \sqcap B) \sqsubseteq \exists r.A$ $\boxed{\text{PROOF}}$ This means that for any inter I and any onto O, we have: $(\exists r.(A \sqcap B))^I \subseteq (\exists r.A)^I := \{d \in \Delta^I \mid \exists e \in \Delta^I \text{ s.t } (d,e) \in r^I \text{ and } e \in A^I\}$. Let's assume that we have an inter I and an object $x \in (\exists r(A \sqcap B))^I = \{d \in \Delta^I \mid \exists e \in \Delta^I \text{ s.t } (d,e) \in r^I \text{ and } e \in (A \sqcap B)^I = A^I \cap B^I\}$. This basically means that there exists an object e s.t $(d,e) \in r^I$ and $e \in A^I \cap B^I$, which means that $e \in A^I$, thus $x \in (\exists r.A)^I$. As this holds for any inter I and any onto O, we can conclude that the subsumption holds.

$\{A \sqsubseteq \exists r.B, B \sqsubseteq C\} \models A \sqsubseteq \exists r.C$ $\boxed{\text{PROOF}}$ Let's assume an arbitrary inter I that satis the TBox T, then we have that: 1) $A \sqsubseteq \exists r.B \Leftrightarrow A^I \subseteq (\exists r.B)^I$. We deduce that from any $x \in A^I$ there is some $e \in B^I$ s.t $(x,e) \in r^I$. 2) $B \sqsubseteq C \Leftrightarrow B^I \subseteq C^I$. We infer that for any $e \in B^I$, $e \in C^I$. Taking T into account the previous info, we need to show that $\forall x \in A^I, \exists e \in C^I \text{ s.t } (d,e) \in r^I$ for $d \in \Delta^I$. Combining both inferences, we can deduce that for any $x \in A^I$, there is some $e \in C^I$ s.t $(x,e) \in r^I$, which is equivalent to stating $x \in (\exists r.C)^I$. Hence I satis $A \sqsubseteq \exists r.C$ and since I arbitrary, this holds for any inter I.

$\exists r.A \sqcap \exists r.B \not\sqsubseteq \exists r.(A \sqcap B)$ $\boxed{\text{PROOF}}$ We aim to see if $\exists r.A \sqcap \ldots$ holds, i.e., if an individual is related to some A via r and some B via r, does that guarantee that they are related to a single indi who is both in A and B via r? Let's assume arbitrary I and object $x \in (\ldots)^I$: (1) $(\exists r.A \sqcap \exists r.B)^I = (\exists r.A)^I \cap (\exists r.B)^I$ (2) $(\exists r.A^I)$ = def (3) $(\exists r.B)^I$ = def (4) $(\exists r.A \sqcap B)^I = \{d \in \Delta^I \mid \exists e \in \Delta^I \text{ s.t } (d,e) \in r^I \text{ and } e \in (A \sqcap B)^I = A^I \cap B^I\}$. We provide a counterexample to show it does not hold. We consider an indi x related via r to 2 distinct individuals e_1 and e_2, where $e_1 \in A^I$, $e_1 \notin B^I$ and $e_2 \notin A^I$ and $e_2 \in B^I$. Therefore, if $x \in (\exists r.A)^I$, then there exists some indi e_1 s.t $(x,e_1) \in r^I$ and $e_1 \in A^I$, similarly if $x \in (\exists r.B)^I$, $\exists e_2 \in \Delta^I$ s.t $(x,e_2) \in r^I$ and $e_2 \in B^I$. Since $e_1 \neq e_2$, there is no single indi e s.t $(x,e) \in r^I$ and $e \in A^I \cap B^I$. Then: $x \in (\exists r.A \sqcap \exists r.B)^I$ but $x \notin (\exists r.(A \sqcap B))^I$. The subsumption doesn't hold.

$\forall r.A \sqcap \forall r.B \sqsubseteq \forall r.(A \sqcap B)$ $\boxed{\text{PROOF}}$ From $x \in (\forall r.A)^I$, for all $e \in \Delta^I$, if $(x,e) \in r^I$ then $e \in A^I$, and analogous for $x \in (\forall r.B)^I$. Combining both statements, $e \in A^I \cap B^I$, therefore $x \in (\forall r.(A \sqcap B)^I)$. The subsumption holds.

TRUE & FALSE STATEMENTS

- OWL and OWL2 are 2 versions of the W3C for onto web language standard: T
- Onto classification is to determine the classes to which each instance belongs: F
- There is no algo that can decide the satisfi of an ALC concept with a complexity less than PSPACE: T
- For 2 ontos $O \subseteq O'$, we know that $Mod(O') \subseteq Mod(O)$: T
- $\exists s(A \sqcap B) = \exists s.A \sqcap \exists s.B$: F
- Each RDF needs to have at least one blank node: F
- There are OWL axioms that can't be represented in Turtle: F
- The algo Csat of deciding satis of a concept A w.r.t an onto O can be used to decide subsumption entailment, i.e $O \models A \sqsubseteq B$: T
- The language ALC is the underlying formal language for OWL2EL, a profile of OWL2 standard: F
- $\forall s(A \sqcap B) = \forall s.A \sqcap \forall s.B$: T
- Once the syntax is formally defined, the output of all tools for querying RDF triples will be the same: F
- If an algo is sound and terminates, then it can guarantee that its output is always correct: F
- A concept C is satis $\Leftrightarrow C \sqsubseteq \top$ holds: F
- The only onto with no models is the empty onto: F
- $\{A \sqsubseteq \exists r.B, B \sqsubseteq C\} \models A \sqsubseteq \forall r.C$: T
- An onto can have no models: T
- We can use SPARQL to decide the satisfiability of an onto and define soundness of a reasoning system without defining semantics: F
- HermiT is a reasoner that can deal with DL language ALC: T
- We can use sparql to check satisfiability w.r.t an onto: F
- If an onto is inconsistent, then each sub-onto is still inconsistent: F
- Sparql is the query language over RDF triples: T
- Without semantics, different tools can give different answers when querying RDF triples: T
- There is a unique way to define semantics for a language: F
- OWL and OWL2 both use model theoretic semantics: T
- We can define completeness of a reasoning system without defining semantics: F
- ALC contains at least one OWL2 profile: F
- An OWL onto can have infinite interpretations: T
- RDF, RDFs, OWL, OWL2 are all W3C standards with well defined formal semantics: T
- There is an incoherent but consistent onto: T

TABLEAU ALGORITHM

① $\forall R.(A \sqcup \exists S.B) \sqcap \forall R.\neg A \sqcap \neg\exists R.(\neg A \sqcap \exists S.B)$

Applying NNF: $\forall R.(A \sqcup \exists S.B) \sqcap \forall R.\neg A \sqcap \forall R.(A \sqcup \forall S.\neg B)(a_0)$

Csat: $\forall R.(A \sqcup \exists S.B) \sqcap \forall R.\neg A \sqcap \forall R.(A \sqcup \forall S.\neg B)(a_0)$
$\Leftrightarrow \forall R.(A \sqcup \exists S.B)(a_0), \forall R.\neg A(a_0), \forall R.(A \sqcup \forall S.\neg B)(a_0)$

As we don't have any relation, the concept is clash-free, complete and done.

② TBox $= \emptyset$ & $C = \sqcap_{0 \leq i \leq 3} \underbrace{\exists R \ldots \exists R}_{i \text{ times}}.(\forall R.B \sqcap \forall R.\neg B)$

$C = (\forall R.B \sqcap \forall R.\neg B) \sqcap \exists R.(\forall R.B \sqcap \forall R.\neg B) \sqcap \exists R\exists R(\forall R.B \sqcap \forall R.\neg B) \sqcap \exists R.\exists R.\exists R.(\forall R.B \sqcap \forall R.\neg B)$

Csat: $C(a_0) \Leftrightarrow (\forall R.B \sqcap \forall R.\neg B)(a_0), \exists R.(\forall R.B \sqcap \forall R.\neg B)(a_0), \exists R\exists R(\forall R.B \sqcap \forall R.\neg B)(a_0), \exists R\exists R\exists R(\forall R.B \sqcap \forall R.\neg B)(a_0)$

(*): $R(a_0,a_1), (\forall R.B \sqcap \forall R.\neg B)(a_1)$

⊖: $\forall R.B(a_0), \forall R.\neg B(a_0) \Leftrightarrow B(a_1), \neg B(a_1) \Rightarrow$ clash $\Rightarrow$ not satis

③ TBox $= \{A \sqsubseteq B \sqcap \exists r.C, B \sqsubseteq \forall r.\neg C\}$; ABox $= \{r(a,b), A(a)\}$

¿Is $O = $ (TBox, ABox) consistent? We know that $A(a)$, $r(a,b)$.

From the first concept: $B \sqcap \exists r.C(a_0) \Leftrightarrow B(a_0), \exists r.C(a_0)$

Then, from $\exists r.C(a_0) \Rightarrow r(a_0,a_1), C(a_1)$. However, $B \sqsubseteq \forall r.\neg C$, and as $B(a_0)$, we have that $\forall r.\neg C(a_0) \Rightarrow \neg C(a_1)$. Clash.

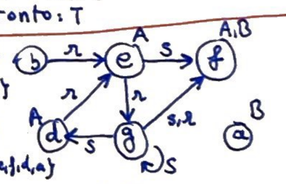
Csat returns "yes" on input $C_0 \Leftrightarrow C_0$ is satisfiable.

Woman(Mary) $\Leftrightarrow$ Mary is a woman.
Mother $\sqsubseteq$ Woman $\Leftrightarrow$ Every mother is a woman.
symmetric(isMarriedTo), isMarriedTo(John, Mary) $\Leftrightarrow$ Mary is John's wife.
Mother $\equiv$ Woman $\sqcap$ Parent $\Leftrightarrow$ Mothers are women who are also parents.
GrandParent $\sqsubseteq$ ($\exists$hasChild.($\exists$hasChild.$\top$)) $\Leftrightarrow$ At least one child of a grandparent has also a child.

A course is evaluated by examinations. An examination can be a midterm exam, a final exam, projects, homework, graded practical labs, presentations or similar items. Each examination is graded on a scale 0-20, with 20 being the maximal grade and 10 being the passing grade. Pierre took the course Semantic Web, a mandatory lecture in a master's program and he got 15 as his final grade.

$\Updownarrow$ Course $\sqsubseteq \exists$evaluatedBy.Examination
Examination $\sqsubseteq$ MidtermExam $\sqcup$ FinalExam $\sqcup$ Project $\sqcup$ Homework $\sqcup$ PracticalLab $\sqcup$ Presentation
Examination $\sqsubseteq \exists$hasScale.{20}
passingGrade = 10   takeCourse(Pierre, SemanticWeb)
highestGrade = 20   MandatoryLecture(SemanticWeb)   hasSWGrade(Pierre, 15)

- $(A \sqcup \neg B)^I = \{e,d,f,b,g\}$
- $(\neg(B \sqcup (\exists s.A)))^I = \{b,d\}$
  $\{f,a\}$ $\{e,g\}$ $\rightarrow \{g\}$
- $\neg\exists r.(A \sqcap B) = \{b,e,f,d,a\}$
- $(\exists s.A)^I = \{e,g\}$
- $(\forall s.A)^I = \{e\}$
- $(\exists s.\neg A)^I = \{g\}$
- $(\forall s.\neg A)^I = \emptyset$



$I = (\Delta^I, \cdot^I)$ where $\Delta^I = \{b,e,f,d,g,a\}$; $A^I = \{d,e,f\}$; $B^I = \{f,a\}$; $r^I = \{(b,e),(d,e),(e,g)\}$, $s^I = \{(e,f),(g,d),(g,f),(g,g)\}$

DEFINITION OF GRAPH

$\exists s.A$	1) Look at relation s pairs 2) check which second element $\in$ A 3) Include 1st element if 2nd $\in$ A
$\forall s.A$	1) Look at relation s pairs 2) If all of the second elem from a given indi are in A then include it, otherwise not

$\forall s.A \Leftrightarrow \neg(\exists s.\neg A)$
$\forall s.\neg A \Leftrightarrow \neg(\exists s.A)$

$\neg(B \sqcup (\exists s.A)) = \neg B^I \sqcap \forall s.\neg A^I = \{b,d,e,g\} \sqcap \{a,b,d,f\} = \{b,d\}$