∞-Categories in Algebraic Geometry
Université Paris–Saclay (Orsay)

Lecture 1. Prelude: Morita Theory

Koszul duality patterns have influenced several recent developments in algebraic geometry, ranging from the classification of formal deformations by Lie algebras and the unobstructedness of Calabi–Yau varieties to purely inseparable Galois theory and derived Galois deformation rings. In this class, we will explain some of these results and review the required ∞-categorical background.

1.1. **Categorical Morita Theory.** Before discussing Koszul duality, which is an inherently higher categorical phenomenon, we will review Morita theory [Mor58], a good toy example which can be treated using only ordinary categories. It is centered around the following simple question:

Question. Given two associative rings R and S, is there an equivalence between the categories of left modules Mod_R and Mod_S?

If such an equivalence exists, then the rings R and S are said to be *Morita equivalent*. Isomorphic rings are clearly Morita equivalent, but the converse need not be true:

Proposition 1.1 (Morita functors)**.** *Let $Q \in \mathrm{Mod}_R$ be a left module over a ring R such that*

(1) Q is finite projective, i.e. a direct summand of $R^{\oplus n}$ for some n;
(2) Q is a generator, which means that the functor $\mathrm{Hom}_R(Q, -)$ is faithful.

Then R and $S = \mathrm{End}_R(Q)^{op}$ are Morita equivalent, which is witnessed by inverse equivalences

$$\widetilde{G} : \mathrm{Mod}_R \to \mathrm{Mod}_S, \quad M \mapsto \mathrm{Hom}_R(Q, M)$$
$$\widetilde{F} : \mathrm{Mod}_S \to \mathrm{Mod}_R, \quad N \mapsto Q \otimes_S N.$$

Before proving this claim, we give a simple exercise:

Exercise 1.2 (Examples of Morita equivalences)**.**

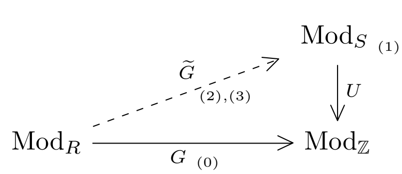
a) Prove directly that for any ring R and any $n > 0$, the ring R is Morita equivalent to $M_n(R)$.
b) Find a ring R and a finite projective generator $Q \in \mathrm{Mod}_R$ such that $S = \mathrm{End}_R(Q)^{op}$ is not a matrix algebra.

We present a categorical proof of Proposition 1.1 which we have learned from [Lur, Section 4.8]. While needlessly abstract, it will generalise well to ∞-categories of chain complexes and serve as good excuse to revise some basic categorical notions.

Our goal is to implement the following strategy:

Strategy 1.3.

(0) Consider the functor $G : \mathrm{Mod}_R \to \mathrm{Mod}_{\mathbb{Z}}$ given by $M \mapsto \mathrm{Hom}_R(Q, M)$;
(1) Construct the associative ring $S = \mathrm{End}_R(Q)^{op}$ from the functor G;
(2) Lift G to a functor $\widetilde{G} : \mathrm{Mod}_R \to \mathrm{Mod}_S$ by exhibiting an S-module structure on each $\mathrm{Hom}_R(Q, M)$;
(3) Show that $\widetilde{G}$ is an equivalence.



1.2. **Properties of Functors.** We begin by reformulating the algebraic conditions imposed in Proposition 1.1 on $Q \in \mathrm{Mod}_R$ in terms of the associated functor $G : \mathrm{Mod}_R \to \mathrm{Mod}_{\mathbb{Z}}$. We treat the "finite" and the "projective" part in (1) separately, and start with the former.

Compactness. The categorical notion of compactness aims to capture the smallness of a given object X by asserting that it cannot be "spread out" arbitrarily.

For example, given a diagram $Y_0 \to Y_1 \to \ldots$, any map from a small object X to the sequential colimit $\mathrm{colim}_i Y_i$ (which we might think of as an increasing union) should factor through some Y_i.

In fact, we will also want to take slightly more general diagrams into account:

Definition 1.4 (Filtered categories)**.** A category I is *filtered* if it is nonempty and

a) any two objects x, y map into a third object z via morphisms $x \to z$, $y \to z$;
b) for all parallel morphisms $f, g : x \rightrightarrows y$ in $\mathcal{C}$, there exists $h : y \to z$ with $h \circ f = h \circ g$.

A *filtered colimit* in a category $\mathcal{C}$ is a colimit over a diagram $D : I \to \mathcal{C}$, where I is filtered.

Exercise 1.5. *Establish the following facts:*

(1) The category $\mathbb{N} = (\bullet \to \bullet \to \ldots)$ is filtered; hence sequential colimits are filtered;
(2) The product of filtered categories is filtered;
(3) The category $\bullet \;\; \bullet$ is not *filtered, and neither is Δ^{op}, the opposite of the category of nonempty finite linearly ordered sets.*

We can explicitly compute filtered colimits in the category of sets:

Exercise 1.6 (Filtered colimits of sets)**.** *Given a diagram $D : I \to \mathrm{Set}$ with I a small filtered category, show that $\mathrm{colim}_{i \in I} D(i)$ is given by the set $\coprod_{i \in I} D(i)/_{\cong}$, where $\cong$ is the equivalence relation identifying $a \in D(i), b \in D(j)$ if there are arrows $f : i \to k$, $g : j \to k$ with $D(f)(a) = D(g)(b)$.*

Exercise 1.7 (Limits of sets)**.** *Given a diagram $D : I \to \mathrm{Set}$ with I small, write down its limit.*

We will often need the following important fact:

Exercise 1.8 (Filtered colimits and finite limits in Set)**.**

a) Given a diagram $D : I \times J \to \mathrm{Set}$ with I a small filtered category and J a category with finitely many objects and morphisms, the following canonical arrow is an isomorphism:

$$\operatorname*{colim}_{i \in I} \left(\lim_{j \in J} D(i,j) \right) \xrightarrow{\cong} \lim_{j \in J} \left(\operatorname*{colim}_{i \in I} D(i,j) \right).$$

b) Show that filtered colimits generally do not commute with limits in Set.
c) Show that in Set^{op}, filtered colimits need not commute with finite limits.

Filtered colimits and finite limits also commute in categories that are sufficiently similar to sets. To make this precise, we need several notions.

Notation 1.9. Given a category I, the *right cone* $I^{\triangleright}$ is obtained from I by adding a new object 1 and a unique morphism from every $i \in I$ to the new object 1.

Definition 1.10. Let I be a category. We say that a functor $F : \mathcal{C} \to \mathcal{D}$ preserves and reflects colimits of shape I if $D^{\triangleright} : I^{\triangleright} \to \mathcal{C}$ is a colimit diagram if and only if this is true for $U \circ D^{\triangleright} : I^{\triangleright} \to \mathcal{D}$. A similar definition applies to limits.

Using that faithful functors reflect isomorphisms (which we establish in Proposition 1.26 below), we can deduce the following basic fact from Exercise 1.8:

Corollary 1.11. *Let $U : \mathcal{C} \to \mathrm{Set}$ be a faithful functor which preserves and reflects finite limits and filtered colimits. Then finite limits commute with filtered colimits in $\mathcal{C}$.*

Exercise 1.12. *Show that for any ring R, the forgetful functor $U : \mathrm{Mod}_R \to \mathrm{Set}$ satisfies the assumptions of Corollary 1.11. Hint: equip the colimit of sets $\mathrm{colim}_{i \in I}(U \circ D)(i)$ constructed in Exercise 1.7 with the structure of an R-module.*

We can now give a categorical notion of smallness:

Definition 1.13. An object X in a locally small category $\mathcal{C}$ is called *compact* if the functor $\mathrm{Map}_{\mathcal{C}}(X, -) : \mathcal{C} \to \mathrm{Set}$ preserves filtered colimits.

Using Exercise 1.8, we can prove an intuitive closure property for compact objects:

Corollary 1.14. *Finite colimits of compact objects in a category $\mathcal{C}$ are compact.*

Proof. For any finite diagram $D : J \to \mathcal{C}$ which admits a colimit in $\mathcal{C}$, we have a natural isomorphism of functors $\mathrm{Map}_{\mathcal{C}}(\mathrm{colim}_{j \in J} D(j), -) \xrightarrow{\cong} \lim_{j \in J} \mathrm{Map}_{\mathcal{C}}(D(j), -)$. For any filtered diagram $D' : I \to \mathcal{C}$, compactness of all $D(j)$ and Exercise 1.8 implies:

$$\begin{aligned} \mathrm{Map}_{\mathcal{C}}(\mathrm{colim}_{j \in J} D(j), \mathrm{colim}_{i \in I} D'(i)) &\cong \lim_{j \in J} \mathrm{colim}_{i \in I} \mathrm{Map}_{\mathcal{C}}(D(j), D'(i)) \\ \cong \mathrm{colim}_{i \in I} \lim_{j \in J} \mathrm{Map}_{\mathcal{C}}(D(j), D'(i)) &\cong \mathrm{colim}_{i \in I} \mathrm{Map}_{\mathcal{C}}(\mathrm{colim}_{j \in J} D(j), D'(i)) \end{aligned}$$

□

Example 1.15 (Compact sets)**.** A set is compact if and only if it is finite.

For the "if" part, we first observe that the set $*$ with one object is compact. As finite sets are finite coproducts of points, Corollary 1.14 shows that they are compact.

To see the "only if" part, let S be an infinite set and consider the category I with objects $\{ x_T \mid T \subset S \text{ finite} \}$ and a unique morphism $x_T \to x_{T'}$ whenever T is contained in T'. An easy check shows that I is filtered, and that S is the colimit of the functor $D : I \to \mathrm{Set}$ given by $x_T \mapsto T$. If S were compact, then $\mathrm{Map}_{\mathrm{Set}}(S, S) \cong \mathrm{colim}_{i \in I} \mathrm{Map}_{\mathrm{Set}}(S, D(i))$ and we could factor the identity map $S \to S$ through a finite subset, which is absurd.

Exercise 1.16 (Compact topological spaces)**.** *Compact objects in the category of topological spaces are finite sets with the discrete topology. We will revisit this example later.*

Example 1.17 (Compact modules)**.** A (left) module M over a ring R is compact if and only if it is finitely presented. The proof is almost identical to Example 1.15.

First observe that R is compact because $\mathrm{Map}_R(R, M) \cong M$ and the forgetful functor $\mathrm{Mod}_R \to \mathrm{Set}$ preserves filtered colimits by Exercise 1.12. Since any finitely presented R-module is an iterated finite colimit of copies of R, the "if" part follows.

For the converse direction, we need that any R-module is a filtered colimit of finitely presented modules; we leave this as an exercise. If M is compact, then we can factor the identity map on M through a finitely presented submodule. This shows that M is a summand of a finitely presented module, and hence finitely presented itself.

We have completed the first step towards the desired reformulation of Proposition 1.1:

Corollary 1.18. *A module $Q \in \mathrm{Mod}_R$ is finitely presented if and only if the functor $G = \mathrm{Map}_{\mathrm{Mod}_R}(Q, -) : \mathrm{Mod}_R \to \mathrm{Mod}_{\mathbb{Z}}$ preserves filtered colimits.*

Remark 1.19. Since the forgetful functor $\mathrm{Mod}_{\mathbb{Z}} \to \mathrm{Set}$ preserves and reflects filtered colimits, this is an instance of Definition 1.13.

Projectivity. We now give a reformulation of the condition that a module $Q \in \mathrm{Mod}_R$ be projective, with an eye towards later higher-categorical generalisations. First, we recall a well-known result in homological algebra:

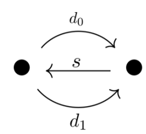
Proposition 1.20. *Given a module $Q \in \mathrm{Mod}_R$, the following are equivalent:*

a) Q is a summand of a free module;
b) The functor $\mathrm{Map}_R(Q, -)$ preserves surjections;
c) The functor $\mathrm{Map}_R(Q, -)$ preserves short exact sequences;
d) The functor $\mathrm{Map}_R(Q, -)$ preserves cokernels.

If these conditions hold, we call the module Q projective.

We will reformulate the "cokernel" condition d) using the following notion:

Definition 1.21. A *reflexive pair* in a category $\mathcal{C}$ is a diagram consisting of two arrows $d_0, d_1 : X_1 \rightrightarrows X_0$ and a common section $s : X_0 \to X_1$ satisfying $f \circ s = g \circ s = \mathrm{id}_{X_0}$. In other words, it is a $\Delta^{op}_{\leq 1}$-indexed diagram, where $\Delta_{\leq 1}$ is the category of nonempty ordered sets of cardinality ≤ 1; we will return to this perspective in the next lectures.



A *reflexive coequaliser* is the colimit of a reflexive pair. Note that this agrees with the coequaliser of the arrows d_0 and d_1.

We also record the following notion:

Definition 1.22. A functor $F : \mathrm{Mod}_R \to \mathrm{Mod}_{\mathbb{Z}}$ is called *additive* if for all M, N, the functor $\mathrm{Map}_R(M, N) \to \mathrm{Map}_{\mathbb{Z}}(FM, FN)$ is a homomorphism of abelian groups.

Condition d) in Proposition 1.20 can be reformulated in terms of reflexive coequalisers:

Proposition 1.23. *An additive functor $F : \mathrm{Mod}_R \to \mathrm{Mod}_{\mathbb{Z}}$ preserves cokernels if and only if it preserves reflexive coequalisers.*

Proof. Assume that F preserves cokernels. The coequaliser of a reflexive pair $A \underset{g}{\overset{f}{\rightleftarrows}} B$ is the cokernel of $A \xrightarrow{f-g} B$. As F is additive, this shows that it preserves reflexive coequalisers. Conversely, assume that F preserves reflexive coequalisers. The cokernel of $A \xrightarrow{f} B$ agrees with the coequaliser of the reflexive pair $A \oplus B \underset{\mathrm{id}_B}{\overset{f+\mathrm{id}_B}{\rightleftarrows}} B$, which implies the claim. ☐

Corollary 1.24. *A module $Q \in \mathrm{Mod}_R$ is projective if and only if the functor $\mathrm{Map}_R(Q, -)$ preserves reflexive coequalisers.*

Exercise 1.25.

a) Prove that the forgetful functor $\mathrm{Mod}_{\mathbb{Z}} \to \mathrm{Set}$ preserves and reflects reflexive coequalisers.
b) Show that this becomes false once we drop the word "reflexive".

Conservativity. Recall that a functor $G : \mathcal{C} \to \mathcal{D}$ is called *conservative* if $f : X \to Y$ is an isomorphism whenever $G(f)$ is one. We can then reformulate condition (2) of Proposition 1.1 by making the following simple observation:

Proposition 1.26. *Any faithful functor* $G : \mathrm{Mod}_R \to \mathrm{Mod}_{\mathbb{Z}}$ *is conservative. Any conservative functor which preserves coequalisers is faithful.*

Proof. First assume that G is faithful. If $G(f)$ is an isomorphism, then it is both an epi- and a monomorphism. Since G is faithful, this implies that f is both an epi- and a monomorphism, which shows that f is an isomorphism since Mod_R is an abelian category.

Conversely, assume that G is a conservative functor which preserves coequalisers. Note that arrows $f, g : A \to B$ are equal if and only if in the coequaliser diagram $A \overset{f}{\underset{g}{\rightrightarrows}} B \xrightarrow{h} C$, the map h is an isomorphism; this condition is preserved and reflected by the functor G. □

Coming back to Proposition 1.1, we can now rephrase algebraic conditions imposed on Q in terms of categorical conditions on the functor $G = \mathrm{Hom}_R(Q, -)$:

Q is finitely presented	$\leftrightsquigarrow$	G preserves filtered colimits, i.e. Q is compact;
Q is projective	$\leftrightsquigarrow$	G preserves reflexive coequalisers;
Q is a generator	$\rightsquigarrow$	G is conservative.

1.3. **Monads and Adjunctions.** To construct the crucial diagram in Strategy 1.3, we will first use that G admits a left adjoint to construct a monad T on $\mathrm{Mod}_{\mathbb{Z}}$, and then identify T-algebras with S-modules. We briefly review the categorical notions appearing in this sentence.

Monads. Monads provide a way of axiomatising algebraic structures that is convenient for certain abstract arguments. We start with a simple example:

Example 1.27 (Groups)**.** Traditionally, groups are defined as sets X with a binary multiplication $(x, y) \mapsto x \cdot y$, a unary inverse $x \mapsto x^{-1}$, and a unit e satisfying various axioms.

We could also choose a less economical approach, and specify many more operations, e.g.

$$(1) \qquad (x_1, x_2, x_3) \mapsto x_1 \cdot x_3^{10} \cdot x_2^{-1}, \quad (x_1, x_2, x_3, x_4) \mapsto x_1^4 \cdot x_2^2 \cdot x_3 \cdot x_4^{-15}, \quad \text{etc.}$$

More precisely, consider the endofunctor $T_{\mathrm{Gp}} : \mathrm{Set} \to \mathrm{Set}$ sending a set X to the set of expressions

$$T_{\mathrm{Gp}}(X) \ := \ \{ \ x_1^{a_1} x_2^{a_2} \dots x_k^{a_k} \ \mid \ k \geq 0, \ x_i \in X, \ a_k \in \mathbb{Z} - \{0\}, \ x_i \neq x_{i+1} \text{ for all } i. \ \}$$

Here the empty word ( ) is considered a valid element of the set $T_{\mathrm{Gp}}(X)$.

In our uneconomical approach to groups, defining all operations as in (1) amounts to specifying a single map $\alpha : T_{\mathrm{Gp}}(X) \to X$ sending a formal expression $x_1^{a_1} x_2^{a_2} \dots x_k^{a_k}$ to the value of the corresponding product $x_1^{a_1} \cdot x_2^{a_2} \cdot \dots \cdot x_k^{a_k}$ in X.

However, not all such maps $\alpha : T_{\mathrm{Gp}}(X) \to X$ define valid group structures on the set X, as we have not yet imposed any of the group axioms. To fix this, we exhibit additional structure on the endofunctor T_{Gp} by specifying the following natural maps for all sets X:

$$\eta_X : X \to T_{\mathrm{Gp}}(X) \qquad\qquad \mu_X : T_{\mathrm{Gp}}(T_{\mathrm{Gp}}(X)) \to T_{\mathrm{Gp}}(X).$$

The first map η_X takes an element $s \in X$ to the corresponding one-letter word in $T_{\mathrm{Gp}}(X)$. The second map μ_X sends a "word of words" $(x_{11}^{a_{11}} \dots x_{1k_1}^{a_{1k_1}})^{b_1} \dots \dots (x_{n1}^{a_{n1}} \dots x_{nk_n}^{a_{nk_n}})^{b_n}$ in $T_{\mathrm{Gp}}(T_{\mathrm{Gp}}(X))$

to the corresponding word in $T_{\mathrm{Gp}}(X)$ given by

$$\underbrace{(x_{11}^{a_{11}} \dots x_{1k_1}^{a_{1k_1}}) \dots (x_{11}^{a_{11}} \dots x_{1k_1}^{a_{1k_1}})}_{b_1} \dots \dots \underbrace{(x_{n1}^{a_{n1}} \dots x_{nk_n}^{a_{nk_n}}) \dots (x_{n1}^{a_{n1}} \dots x_{nk_n}^{a_{nk_n}})}_{b_n}$$

Here, we have implicitly simplified this word by reducing subwords of the form $x^a x^b$ to x^{a+b}.

Exercise 1.28. *The maps η_X and μ_X are natural in X and satisfy the following identities:*

$$\mu_X \circ T_{\mathrm{Gp}}(\mu_X) \cong \mu_X \circ \mu_{T_{\mathrm{Gp}}(X)}, \qquad \mu_X \circ \eta_{T_{\mathrm{Gp}}(X)} = \mathrm{id}_{T_{\mathrm{Gp}}(X)} = \mu_X \circ T_{\mathrm{Gp}}(\eta_X).$$

Using the natural transformations η and μ, we can now formulate a condition for when a map $\alpha : T_{\mathrm{Gp}}(X) \to X$ defines a group structure on X:

Exercise 1.29. *Given a map $\alpha : T_{\mathrm{Gp}}(X) \to X$, the operations $(x, y) \mapsto \alpha(xy)$, $x \mapsto \alpha(x^{-1})$, $e = \alpha(\ )$ define a group structure on X if and only if $\alpha \circ \eta_X = \mathrm{id}_X$ and $\alpha \circ \mu_X = \alpha \circ T_{\mathrm{Gp}}(\alpha)$.*

We therefore obtain a second definition of what a group is, namely a set X together with a map of sets $T_{\mathrm{Gp}}(X) \to X$ satisfying $\alpha \circ \eta_X = \mathrm{id}_X$ and $\alpha \circ \mu_X = \alpha \circ T_{\mathrm{Gp}}(\alpha)$.

Definitions of this kind can also be given for most other algebraic structures of interest (like modules, rings, Lie algebras, ...). We therefore axiomatise this situation:

Definition 1.30 (Monads)**.** A *monad* on a category $\mathcal{C}$ is an associative algebra object in the monoidal category $\mathrm{End}(\mathcal{C})$ of endofunctors (with the composition product $\circ$).

Concretely, this means that a monad is an endofunctor $T : \mathcal{C} \to \mathcal{C}$ equipped with natural transformations $\mathrm{id}_{\mathcal{C}} \to T$ and $\mu : T \circ T \to T$ such that the following two diagrams commute:

$$\begin{array}{ccc} T \circ T \circ T & \xrightarrow{T(\mu)} & T \circ T \\ {\scriptstyle \mu_T}\downarrow & & \downarrow{\scriptstyle \mu} \\ T \circ T & \xrightarrow[\mu]{} & T \end{array} \qquad \begin{array}{ccc} T & \xrightarrow{\eta_T} & TT \\ {\scriptstyle T(\eta)}\downarrow & \searrow^{\mathrm{id}} & \downarrow{\scriptstyle \mu} \\ T^2 & \xrightarrow[\mu]{} & T \end{array}$$

Definition 1.31 (Algebras over monads)**.** An *algebra* over a monad T on $\mathcal{C}$ is a T-module object in the $\mathrm{End}(\mathcal{C})$-tensored category $\mathcal{C}$. Concretely, this means that an algebra is a pair $(A \in \mathcal{C}, \alpha : T(A) \to A)$ for which the following two diagrams commute:

$$\begin{array}{ccc} A & \xrightarrow{\eta_A} & T(A) \\ & \searrow_{\mathrm{id}_A} & \downarrow{\scriptstyle \alpha} \\ & & A \end{array} \qquad \begin{array}{ccc} T(T(A)) & \xrightarrow{T(\alpha)} & T(A) \\ {\scriptstyle \mu_A}\downarrow & & \downarrow{\scriptstyle \alpha} \\ T(A) & \xrightarrow[\alpha]{} & A \end{array}$$

We write $\mathrm{Alg}_T(\mathcal{C})$ for the category of T-algebras in $\mathcal{C}$.

In Example 1.27, we constructed a monad T_{Gp} acting on Set whose category of algebras $\mathrm{Alg}_{T_{\mathrm{Gp}}}(\mathrm{Set})$ is equivalent to the category of groups. We can construct similar monads for other algebraic structures:

Exercise 1.32.

a) Define a monad T_{Ab} on the category of sets Set *such that* $\mathrm{Alg}_{T_{\mathrm{Ab}}}(\mathrm{Set})$ *is equivalent to the category* $\mathrm{Ab} = \mathrm{Mod}_{\mathbb{Z}}$ *of abelian groups.*

b) Define a monad T_{Ring} on the category Ab *such that* $\mathrm{Alg}_{T_{\mathrm{Ring}}}(\mathrm{Ab})$ *is the category of rings.*

c) Given a ring R, define a monad T_{Ring} on Ab *whose category of algebras is equivalent to the category of (left) R-modules.*

Adjunctions. In Example 1.27, we have adopted the perspective that the monad T_{Gp} can be used as a tool for defining the notion of a group.

We could also reverse this logic and try to define the monad T_{Gp} assuming that we already know what a group is. To this end, recall the following standard notion from category theory (which we will later generalise to higher categories):

Definition 1.33 (Adjunctions)**.** An adjunction consists of functors $F : \mathcal{C} \leftrightarrows \mathcal{D} : G$ together with natural transformations $\eta : \mathrm{id}_{\mathcal{C}} \to GF$ (the "unit"), $\epsilon : FG \to \mathrm{id}_{\mathcal{D}}$ (the "counit") for which the following diagrams commute:

$$\begin{array}{ccc} F & \xrightarrow{F(\eta)} & FGF \\ & \searrow^{\mathrm{id}_F} & \downarrow{\scriptstyle \epsilon_F} \\ & & F \end{array} \qquad\qquad \begin{array}{ccc} G & \xrightarrow{\eta_G} & GFG \\ & \searrow^{\mathrm{id}_G} & \downarrow{\scriptstyle G(\epsilon)} \\ & & G \end{array}$$

The functor F is called the left adjoint, whereas G is called a right adjoint; we write $F \dashv G$.

Remark 1.34. Fix an adjunction (F, G, η, ϵ) as in Definition 1.33. For any pair of objects $X \in \mathcal{C}$ and $Y \in \mathcal{D}$, we obtain natural isomorphisms $\mathrm{Map}_{\mathcal{D}}(FX, Y) \cong \mathrm{Map}_{\mathcal{C}}(X, GY)$. Indeed, given $f : FX \to Y$ in $\mathcal{D}$, we attach the map $\overline{f} : X \to GY$ defined by $\overline{f} = Gf \circ \eta_X$. Conversely, to a map $g : X \to GY$, we attach the map $\overline{g} = \epsilon_Y \circ Fg : FX \to Y$. In fact, specifying natural isomorphisms $\mathrm{Map}_{\mathcal{D}}(FX, Y) \cong \mathrm{Map}_{\mathcal{C}}(X, GY)$ leads to an equivalent definition of adjunctions

Example 3.1 (continued)**.** *There is a free-forgetful adjunction* Free : Set $\leftrightarrows$ Gp : Forget *between the category of sets and the category of groups. The right adjoint* Forget *sends a group to its underlying set, and the left adjoint* Free *builds the free group on a given set. The unit* $\eta_X : X \to \mathrm{Forget}(\mathrm{Free}(X))$ *embeds a set X into the free group generated by X. The counit* $\epsilon_G : \mathrm{Free}(\mathrm{Forget}(G)) \to G$ *takes a formal product* $g_1^{a_1} \dots g_n^{a_n}$ *in the free group on the set G and computes the corresponding product* $g_1^{a_1} \cdot \ldots \cdot g_n^{a_n}$ *in the group G.*

We note that the endofunctor $T_{\mathrm{Gp}} : \mathrm{Set} \to \mathrm{Set}$ *defined above is equal to the composite* $\mathrm{Forget} \circ \mathrm{Free}$*. The transformation* $\mathrm{id}_{\mathrm{Set}} \to T_{\mathrm{Gp}}$ *agrees with the unit η of the adjunction, and the monad multiplication* $\mu : T_{\mathrm{Gp}} \circ T_{\mathrm{Gp}} \to T_{\mathrm{Gp}}$ *is given by* $G\epsilon_F : GFGF \to GF$*.*

The functor $\mathrm{Gp} \to \mathrm{Alg}_{T_{\mathrm{Gp}}}(\mathrm{Set})$ *sending a group G to the T_{Gp}-algebra*

$$\Big(\mathrm{Forget}(G)\ ,\ T_{\mathrm{Gp}}(\mathrm{Forget}(G)) \xrightarrow{\mathrm{Forget}(\epsilon_G)} \mathrm{Forget}(G)\Big)$$

gives the equivalence between groups and T_{Gp}-algebras mentioned above.

Indeed, we obtain a monad for every adjunction:

Exercise 1.35 (Monads from adjunctions)**.** *Given an adjunction* $F : \mathcal{C} \leftrightarrows \mathcal{D} : G$ *with unit* $\eta : \mathrm{id}_{\mathcal{C}} \to GF$ *and counit* $\epsilon : FG \to \mathrm{id}_{\mathcal{D}}$*, show that the endofunctor $T = GF$ is equipped with the structure of a monad with unit* $\eta : \mathrm{id}_{\mathcal{C}} \to GF$ *and multiplication* $G\epsilon_F : T \circ T \to T$*.*

Exercise 1.36. *Given a monad T on a category $\mathcal{C}$, consider the functor* $\mathrm{Free}_T : \mathcal{C} \to \mathrm{Alg}_T(\mathcal{C})$ *sending an object $X \in \mathcal{C}$ to the T-algebra* $(TX,\ T(T(X)) \xrightarrow{\mu_X} T(X))$*.*

a) Prove that Free_T *is a left adjoint to the forgetful functor* $\mathrm{Forget}_T : \mathrm{Alg}_T(\mathcal{C}) \to \mathcal{C}$*.*
b) Verify that the adjunction $\mathrm{Free}_T \dashv \mathrm{Forget}_T$ *induces the monad T.*

This implies the interesting fact that any monad is induced by an adjunction.

Notation 1.37. We will usually denote the free T-algebra on an object $X \in \mathcal{C}$ by $T(X)$ instead of $\mathrm{Free}_T(X)$. Moreover, we will often drop the functor Forget_T from our notation.

If $T = GF$ is a monad obtained from an adjunction $F \dashv G$, we always obtain a functor

$$\widetilde{G} : \mathcal{D} \to \mathrm{Alg}_T(\mathcal{C})$$

sending an object $X \in \mathcal{D}$ to the T-algebra $(G(X)\ ,\ T(G(X)) \xrightarrow{G(\epsilon_X)} G(X))$.

Coming back to Proposition 1.1, we can now give a purely categorical construction of the category Mod_S and the functor $\widetilde{G} : \mathrm{Mod}_R \to \mathrm{Mod}_S$ for $S = \mathrm{End}_R(Q)^{op}$, as desired.

Observation 1.38. *The functor* $G = \mathrm{Map}_R(Q, -)$ *admits a left adjoint given by* $F = Q \otimes (-)$.

This tensor-hom-adjunction

$$Q \otimes (-) : \mathrm{Mod}_{\mathbb{Z}} \leftrightarrows \mathrm{Mod}_R : \mathrm{Map}_R(Q, -)$$

induces a monad T_Q on $\mathrm{Ab} \cong \mathrm{Mod}_{\mathbb{Z}}$ which sends M to $\mathrm{Map}_R(Q, Q \otimes M)$. Hence $T_Q(\mathbb{Z}) = \mathrm{End}_R(Q)$.

In fact, we can use the conditions on Q to identify the endofunctor T more explicitly.

Observation 1.39. *The functor* $G = \mathrm{Map}_R(Q, -)$ *preserves biproducts.*

As $G = \mathrm{Map}_R(Q, -)$ also preserves filtered colimits and reflexive coequalisers, it must preserve small colimits. As this is also true for the left adjoint $Q \otimes (-)$, we deduce that the monad $T_Q : \mathrm{Ab} \to \mathrm{Ab}$ preserves small colimits.

Exercise 1.40.

a) *Given two rings* R_1, R_2*, show that a functor* $\mathrm{Mod}_{R_1} \to \mathrm{Mod}_{R_2}$ *is of the form* $M \mapsto B \otimes_{R_1} M$ *for some* (R_2, R_1)*-bimodule* B *if and only if it is right exact and preserves coproducts (this is known as the Eilenberg-Watts theorem).*

b) *Identify* $\mathrm{Alg}_{T_Q}(\mathrm{Ab})$ *with the category of left modules* Mod_S *over the ring* $S = \mathrm{End}_R(Q)^{op}$.

1.4. **The Barr-Beck Theorem.** To prove Proposition 1.1, it remains to show that the induced functor $\widetilde{G} \to \mathrm{Mod}_S \cong \mathrm{Alg}_{T_Q}$ is an equivalence. We will deduce this from the important Barr–Beck theorem, which we will now review. First, let us introduce some terminology:

Definition 1.41. An adjunction $F \dashv G$ with associated monad T is *monadic* if the induced functor $\widetilde{G} : \mathcal{D} \to \mathrm{Alg}_T(\mathcal{C})$ is an equivalence.

In the case of groups, we have seen in Example 1.27 that the forgetful-free adjunction is monadic, thereby giving an alternative definition of groups as T_{Gp}-algebras.

However, not all adjunctions share this desirable property:

Exercise 1.42 (A non-monadic adjunction)**.** *Consider the adjunction* $F : \mathrm{Set} \rightleftarrows \mathrm{Top} : G$ *between sets and topological spaces whose right adjoint* G *sends a space to its underlying set of points, and whose left adjoint* F *equips a set with the discrete topology.*

Show that this adjunction is not monadic. Hint: what does G *do to isomorphisms?*

The Barr-Beck theorem establishes a simple criterion for when an adjunction is monadic:

Theorem 1.43 (Barr-Beck theorem, crude version)**.**
Assume that an adjunction $F : \mathcal{C} \leftrightarrows \mathcal{D} : G$ satisfies the following two properties:
a) $\mathcal{D}$ admits and G preserves reflexive coequalisers;
b) G is conservative (i.e. reflects isomorphisms).
Then $(F \dashv G)$ is monadic, i.e. $\widetilde{G} : \mathcal{D} \xrightarrow{\cong} \mathrm{Alg}_T(\mathcal{C})$ is an equivalence.

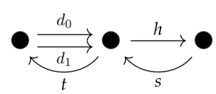
In Definition 1.21, we have introduced the notion of "reflexive coequaliser". To prove Theorem 1.43, we will also need a second notion of coequaliser, which looks similar, but is in fact quite different:

Definition 1.44 (Split coequaliser)**.** Two parallel arrows $d_0, d_1 : X_1 \rightrightarrows X_0$ in a category $\mathcal{C}$ are called a *split pair* if there exist arrows

$$h : X_0 \to X_{-1}, \qquad s : X_{-1} \to X_0, \qquad t : X_0 \to X_1$$

satisfying the following identities:

$$hd_0 = hd_1 \qquad hs = \mathrm{id}_{X_{-1}} \qquad d_0 t = \mathrm{id}_{X_0} \qquad d_1 t = sh$$



Exercise 1.45. *Show that in the situation of Definition 1.44, $X_1 \rightrightarrows X_0 \to X_{-1}$ is a coequaliser. Deduce that it is preserved by any functor – we call this an absolute colimit.*

Using split coequalisers, we can build canonical free resolutions of algebras over monads:

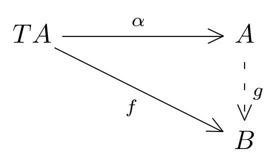
Proposition 1.46 (Free resolutions)**.** *Fix a monad T on a category $\mathcal{C}$ and a T-algebra specified by $(A, \alpha : T(A) \to A)$. The following diagram of T-algebras is a coequaliser in $\mathrm{Alg}_T(\mathcal{C})$:*

$$T(T(A)) \underset{\mu_A}{\overset{T(\alpha)}{\rightrightarrows}} T(A) \xrightarrow{\alpha} A \tag{2}$$

Here, we have used the free functor $\mathcal{C} \to \mathrm{Alg}_T(\mathcal{C})$ from Exercise 1.36 (using Notation 1.37), which sends an object $X \in \mathcal{C}$ to the free T-algebra $(T(X),\ T(T(X)) \xrightarrow{\mu_X} T(X))$ on X.

Proof. Observe that after applying the forgetful functor $\mathrm{Alg}_T(\mathcal{C}) \to \mathcal{C}$, the above diagram is part of a split coequaliser with maps $s = \eta_A : A \to T(A)$ and $t = \eta_{T(A)} : T(A) \to T(T(A))$.

To verify that (2) is also a coequaliser in $\mathrm{Alg}_T(\mathcal{C})$, assume we are given a T-algebra $(B, \beta : T(B) \to B)$ together with a map of T-algebras $f : TA \to B$ with $f \circ T(\alpha) = f \circ \mu_A$. By Exercise 1.45, there is a unique $g = f \circ \eta_A$ in $\mathcal{C}$ such that the following triangle commutes:



Hence, it suffices to check that g is a map of T-algebras, which follows from the computation

$$\beta \circ Tf \circ T(\eta_A) = f \circ \mu_A \circ T(\eta_A) = f = f \circ \mu_A \circ \eta_{TA} = f \circ T(\alpha) \circ \eta_{TA} = (f \circ \eta_A) \circ \alpha.$$

We have used that f is a map of T-algebras, the monad axioms for T, and the naturality of η. $\square$

With these free resolutions at our disposal, we can now prove the Barr-Beck theorem.

Proof of Theorem 1.43. We proceed in three main steps.

Step 1: Left adjoint $\widetilde{F}$ to $\widetilde{G}$. We have a commuting triangle

$$\begin{array}{ccc} & & \mathrm{Alg}_T(\mathcal{C}) \\ & \overset{\widetilde{G}}{\nearrow} & \downarrow{\scriptstyle \mathrm{Forget}_T} \\ \mathcal{D} & \xrightarrow[G]{} & \mathcal{C} \end{array}$$

where both G and Forget_T admit left adjoints (cf. Exercise 1.36).

As left adjoints of commuting right adjoints commute, we know that *if* $\widetilde{G}$ admits a left adjoint $\widetilde{F}$, then its value on free T-algebras must be given by $\widetilde{F}(T(X)) = F(X)$.

Since left adjoints also preserve small colimits, Proposition 1.46 motivates us to define the value of $\widetilde{F}$ on a general T-algebra (A, α) as the following coequaliser in $\mathcal{D}$:

$$F(T(A)) \;\underset{\epsilon_{FA}}{\overset{F(\alpha)}{\rightrightarrows}}\; F(A) \xrightarrow{\theta} \widetilde{F}(A) \tag{3}$$

This makes sense as $F(T(A)) \underset{\epsilon_{FA}}{\overset{F(\alpha)}{\rightrightarrows}} F(A)$ is a reflexive pair in $\mathcal{D}$ with common section $F\eta_A$. One easily extends this definition to morphisms of T-algebras.

To verify that $\widetilde{F}$ is indeed left adjoint to $\widetilde{G}$, we make the following computation:

$$\begin{array}{c} \widetilde{F}(A,\alpha) \to B \\ \hline FA \xrightarrow{f} B \text{ s.t. } f \circ F(\alpha) = f \circ \epsilon_{FA} \\ \hline A \xrightarrow{\overline{f}} B \text{ s.t. } \overline{f} \circ \alpha = G(\epsilon_B)G(F\overline{f}) \\ \hline (A,\alpha) \to \widetilde{G}(B) = (GB, G\epsilon_B). \end{array}$$

In the second step, we have used that $\overline{f} \circ \alpha = \overline{f \circ F(\alpha)} = \overline{f \circ \epsilon_{FA}} \overset{3)}{=} G(f) \overset{4)}{=} G(\epsilon_B)G(F(\overline{f}))$. Here $\overline{(\;)}$ denotes the adjoint bijection on morphisms introduced in Remark 1.34. The first two equalities are straightforward; equalities 3) and 4) follow from the commutative diagrams

$$3)\quad \begin{array}{ccc} GFA & \longrightarrow & GB \\ {\scriptstyle \eta_{GFA}}\downarrow & \overset{\mathrm{id}}{\searrow} & \uparrow{\scriptstyle Gf} \\ GFGFA & \xrightarrow[G\epsilon_{FA}]{} & GFA \end{array} \qquad\qquad 4)\quad \begin{array}{ccc} FA & \xrightarrow{F\overline{f}} & FGB \\ & {\scriptstyle f}\searrow & \downarrow{\scriptstyle \epsilon_B} \\ & & B \end{array} .$$

Step 2: The unit $\mathrm{id}_{\mathrm{Alg}_T(\mathcal{C})} \to \widetilde{F} \circ \widetilde{G}$ is an equivalence.

Given $(A, \alpha) \in \mathrm{Alg}_T(\mathcal{C})$, we have a reflexive coequaliser $F(T(A)) \underset{\epsilon_{FA}}{\overset{F(\alpha)}{\rightrightarrows}} F(A) \xrightarrow{\theta} \widetilde{F}(A)$.

Using that G preserves reflexive coequalisers, we obtain another coequaliser diagram

$$GF(GF(A)) \;\underset{G\epsilon_{FA}}{\overset{GF(\alpha)}{\rightrightarrows}}\; GF(A) \xrightarrow{G\theta} G\widetilde{F}(A)$$

As in the proof of *Proposition* 1.46, the following diagram admits a splitting:

$$GF(GF(A)) \underset{G\epsilon_{FA}}{\overset{GF(\alpha)}{\rightrightarrows}} GF(A) \xrightarrow{\alpha} A$$

Having computed the coequaliser of $GF(GF(A)) \underset{G\epsilon_{FA}}{\overset{GF(\alpha)}{\rightrightarrows}} GF(A)$ in two ways, we obtain an isomorphism

$$\begin{array}{ccc} GFA & \xrightarrow{G\theta} & G\widetilde{F}(A,\alpha) \\ & \searrow^{\alpha} & \downarrow \cong \\ & & A \end{array}$$

We can therefore identify A with $G\widetilde{F}(A,\alpha)$.

Next, we check that $G\epsilon_{\widetilde{F}(A,\alpha)} = \alpha$. Since $\alpha = G\theta$, it suffices to check that $\epsilon_{\widetilde{F}(A,\alpha)} = \theta$. This follows from the following computation:

$$\theta = \theta \circ F\alpha \circ F\eta_A = \theta \circ \epsilon_{FA} \circ F\eta_A = \epsilon_{\widetilde{F}(A,\alpha)} \circ FG(\theta) \circ F\eta_A = \epsilon_{\widetilde{F}(A,\alpha)}$$

In the first and last step, we used the algebra axiom for (A,α), in the second the adjunction axiom relating unit and counit, in the third a naturality square for ϵ.

Altogether, we have verified that $\widetilde{G}(\widetilde{F}(A,\alpha)) = (G\widetilde{F}(A,\alpha), G\epsilon_{\widetilde{F}(A,\alpha)}) \cong (A,\alpha)$.

Step 3: The counit $\widetilde{G} \circ \widetilde{F} \to \mathrm{id}_{\mathcal{D}}$ *is an equivalence.*
By definition, we have a coequaliser diagram computing $\widetilde{F}(\widetilde{G}(B))$:

(4)
$$FGFGB \underset{\epsilon_{FGB}}{\overset{FG\epsilon_B}{\rightrightarrows}} FGB \xrightarrow{\theta} \widetilde{F}(\widetilde{G}(B))$$

By the universal property, the map $\epsilon_B : FGB \to B$ induces a map $\tau : \widetilde{F}(\widetilde{G}(B)) \to B$.

Applying the functor G to the entire situation, we obtain a diagram

$$\begin{array}{ccccc} GFGFGB & \underset{G\epsilon_{FGB}}{\overset{GFG\epsilon_B}{\rightrightarrows}} & GFGB & \longrightarrow & G\widetilde{F}(\widetilde{G}(B)) \\ & & & \searrow & \downarrow \\ & & & & GB \end{array}$$

The top line is a coequaliser as G preserves reflexive coequalisers. The diagram

$$GFGFGB \underset{G\epsilon_{FGB}}{\overset{GFG\epsilon_B}{\rightrightarrows}} GFGB \to GB$$

is a split coequaliser (cf. Proposition 1.46). Together, these facts imply that the map $G\widetilde{F}(\widetilde{G}(B)) \to GB$ is an isomorphism, which shows that $\widetilde{F}(\widetilde{G}(B)) \cong B$ as G is conservative. □

We have almost proven a sharper version of the Barr-Beck theorem. To state it, we need a new notion:

Definition 1.47. Given a functor $G : \mathcal{D} \to \mathcal{C}$, a parallel pair $d_0, d_1 : X_1 \rightrightarrows X_0$ is said to be *G-split* if $G(d_0), G(d_1) : X_1 \rightrightarrows X_0$ is a split pair in the sense of Definition 1.44.

We can now state the desired refinement:

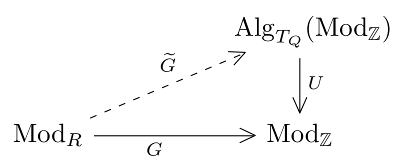
Theorem 1.48 (Barr-Beck theorem, precise version)**.**
An adjunction $F : \mathcal{C} \leftrightarrows \mathcal{D} : G$ is monadic if and only if it has the following two properties:

a) $\mathcal{D}$ admits and G preserves coequalisers of G-split pairs; this means that whenever a pair $d_0, d_1 : X_1 \rightrightarrows X_0$ has the property that $G(X_1), G(X_0) : G(X_1) \rightrightarrows G(X_0)$ is part of a split coequaliser diagram, then $d_0, d_1 : X_1 \rightrightarrows X_0$ admits a colimit, which G preserves.

b) G is conservative (i.e. reflects isomorphisms).

Exercise 1.49. *Taking inspiration from the proof of the crude Barr-Beck Theorem 1.43, prove Theorem 1.48.*

1.5. **Conclusion.** To conclude this lecture, we now give the desired categorical proof of Proposition 1.1. By Observation 1.38, the functor $G = \mathrm{Map}(Q, -) : \mathrm{Mod}_R \to \mathrm{Mod}_{\mathbb{Z}}$ admits a left adjoint $F = Q \otimes (-)$. Writing T_Q for the associated monad on $\mathrm{Mod}_{\mathbb{Z}}$, we obtain a canonical diagram



The functor G preserves biproducts by Observation 1.39, filtered colimits by Corollary 1.18, and reflexive coequalisers by Corollary 1.24. This shows that G and therefore also T_Q preserves small colimits, which allows us to identify $\mathrm{Alg}_{T_Q}(\mathrm{Ab})$ with the category of left $\mathrm{End}_R(Q)^{op}$-modules as in Exercise 1.40. Since G is also conservative by Proposition 1.26, we can apply the crude Barr-Beck theorem Theorem 1.43 to conclude that $\widetilde{G}$ is an equivalence.

Exercise 1.50. *Deduce that all Morita equivalences are realised by the construction in Proposition 1.1, and make this statement precise.*

References

[Lur] Jacob Lurie. Higher algebra. *Preprint from the author's web page.*

[Mor58] Kiiti Morita. Duality for modules and its applications to the theory of rings with minimum condition. *Science Reports of the Tokyo Kyoiku Daigaku, Section A*, 6(150):83–142, 1958.