

Software Design: Modeling with UML

System Analysis and Design

Class 03

Mustafa Hasan



Model

A partial abstract representation of a real-world system

An inexpensive way to analyze, communicate, test, and document our understanding of the system

Types of Models

Computational

- Computer simulations representing time-varying behavior of a system

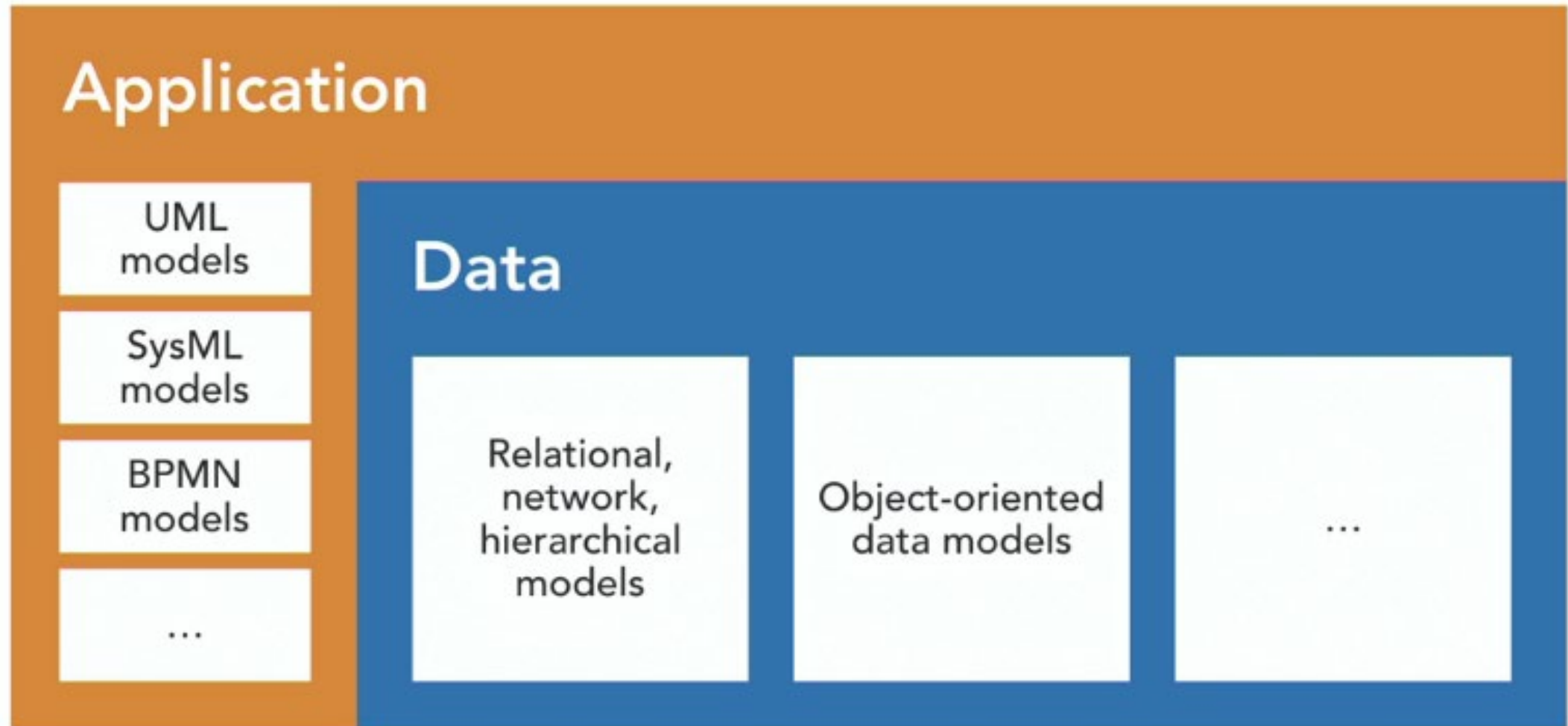
Analytical

- Mathematical models of relationships among variables in a system

Nonanalytical/descriptive

- Describe components and their relationships in a system

Models of Software





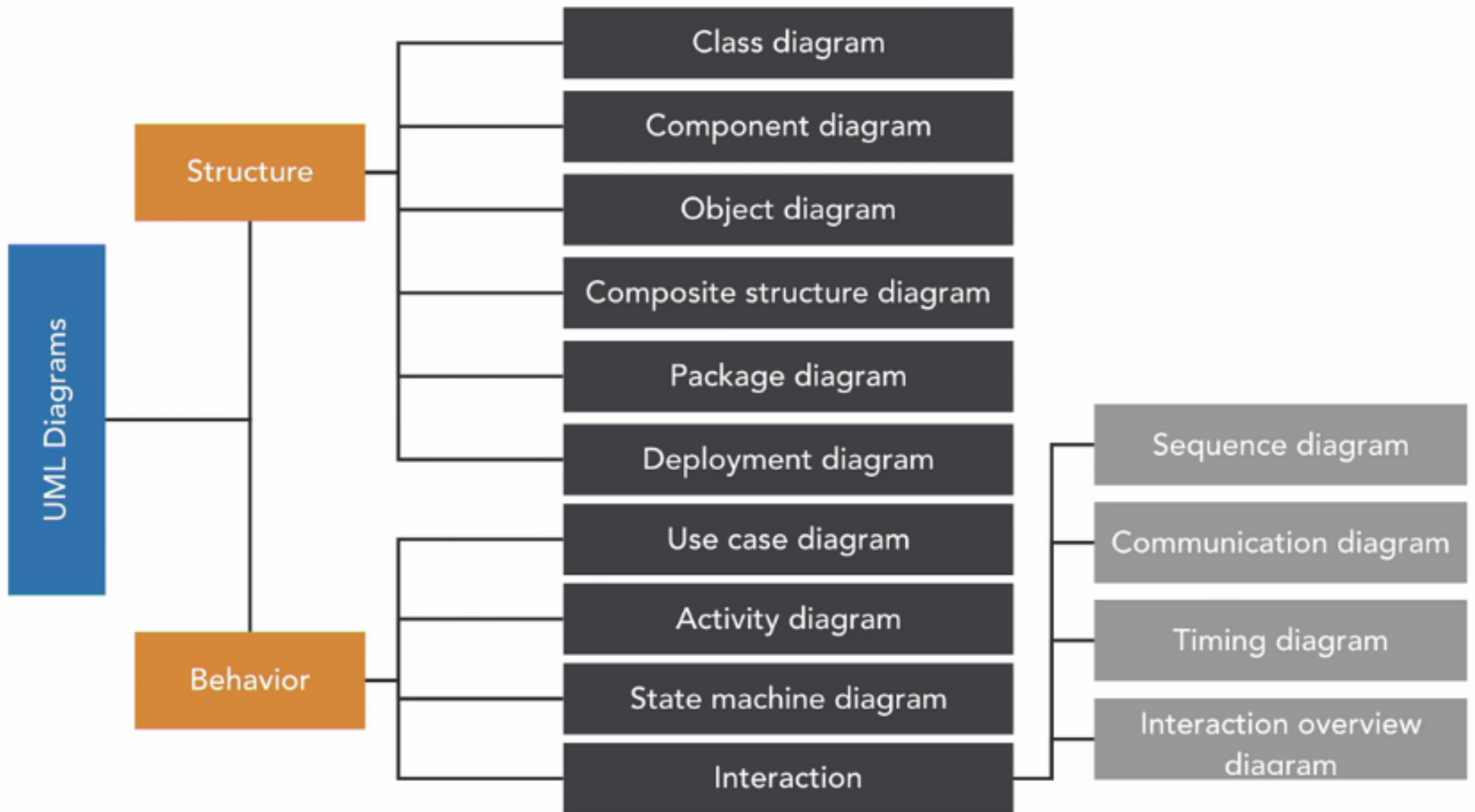
Unified Modeling Language

A family of graphical notations to describe and design software systems, especially those using an object-oriented approach

|

Based on standards controlled by Object Management Group (OMG)

Releases: UML 1 (1997), UML 2 (2005), UML 2.5 (2015)



1

Structure

Represents static view
of the system and
its components

2

Behavior

Represents dynamic
view of the system and
its components

3

Interaction

Represents interaction

- Among components of the system
- Between system and external actors

Important Considerations

- **Model selectively:** you need not (and should not) draw all the models to develop a system
- **Model collaboratively:** use models to think, share, learn, and understand together with your team
- **Model smartly:** start rough and refine it as needed, making it as a long-term asset for the team



CASE Tools

- Computer-aided software engineering tools help in various tasks throughout the software development life cycle
- Some key functions of CASE tools: modeling, code generation, reverse engineering, analyzing code complexity, and other metrics

Name	Creator	Platform / OS	First public release	Latest stable release	Open source
NetBeans ²¹	Oracle Corporation	Windows, macOS, Linux, Unix	1999	2013-02-21	Yes
Borland Together	Borland	Cross-platform (Java)	unknown	2008	No
CasaComplete	Berko Software	Windows	2004	2013-04	No
ConnectOne PRO	CS Onebox	Windows, macOS	1993	2010 v4.6	No
Eclipse Max	EclipseSoft	Windows, Linux, macOS	2004	2010-03	No
Enterprise Architect	Sparx Systems	Windows (supports Linux and macOS installation)	2000	2019-03-08	No
MagiDraw	No Magi	Cross-platform (Java)	1999	2017-02-20 (v18.5)	No
Microsoft Visio	Microsoft	Windows	1993	2019 (v10.0)	No
MyUMLite	Gensoft	Windows, Linux	2003 ²²	unknown	No
PowerDesigner for UML	Centsoftware	Cross-platform (Java)	unknown	2008	No
PowerDesigner	Sybase	Windows	1999	2018	No
Prosa UML Modeler	neofit Oy	Windows	1999	2013-10-19	No
			1999	2019-07-01 (8.1.3)	No
			unknown	2013-05-15	No
		OS, Linux	2009-11-01	2019-06-17	No
			2009-05	2019-10-17	No
		OS, Linux, Solaris	2008-12	2014-10-07	No
		Windows, macOS, Linux	2011-01-12	2018-08-18	No
Visual Paradigm for UML	Visual Paradigm Int'l Ltd.	Cross-platform (Java)	2002-09-20	2019-11-28	No
Caseo	Caseo	Windows 7+, Mac OS X	October 2010	July 2018	No
Gitfy	Gitfy	Chrome, Safari, Firefox, Internet Explorer 9+	2008-06-01	2015-01 (v 5.1)	No
		Linux (RHEL)	2009-09-26 ²³	2019-02-01	No
		Java	2009-10-19	2019-04-29	No
		Java	1997-02	2019-06-27	No
		Java	1999-04	2011-12-18 ²⁴	Yes
		Java	unknown	2010-06-20	Yes
Payara	Commissariat à l'Energie Atomique, Apsa Origin	Windows, Linux, macOS (Java)	2013-04-27	2019-12	Yes
UML Designer	Oban	Windows, macOS, Linux	2012	2017-04-16	Yes
Eclipse UML2 Tools ²⁵	Eclipse Foundation	Cross-platform (Java)	2007	2009-06-19	Yes
gUML	gUMLs GmbH	Windows, macOS, Linux, Unix	unknown	2019-07-18 ²⁶	No
SCUML	Bruno Payle	Cross-platform	2005-02-28	2019-10-23	No
		Windows, macOS X	2002	2019-02-07	No
		Java	unknown	unknown	No
			2008	2010-04-00	No
		Java	20047	2012-07-06	Yes
		OS, Linux, Unix	2009-10-18	2011-06-06	Yes
		Java	2002-02	2009-11-04	Yes
PlantUML	Arnaud Roques	Cross-platform (Java)	1999-04-30	2017-04-03	Yes
Umbrella UML Modeler	Umbrella Team	Unix-like, Windows	2009-09-09	2019-09-29	Yes
UMLet	The UMLet Team	Windows, macOS, Linux	2009-11-26 ²⁷	2019-11-24 ²⁸	Yes
		Windows, macOS, Linux, macOS	2008	2018-04-17 (3.8.1)	Yes
			2011-12-16	2017-08-14 ²⁹	Yes
		Linux	unknown	unknown	No
			Early 1990s	2019-09-18	No
			2004-10-12	2009-09	No
		Java/Eclipse	2008	2019-02-19	Yes
Name	Creator	Platform / OS	First public release	Latest stable release	Open source

Visual Paradigm

Microsoft Visio

PlantUML

IBM Rational Rose

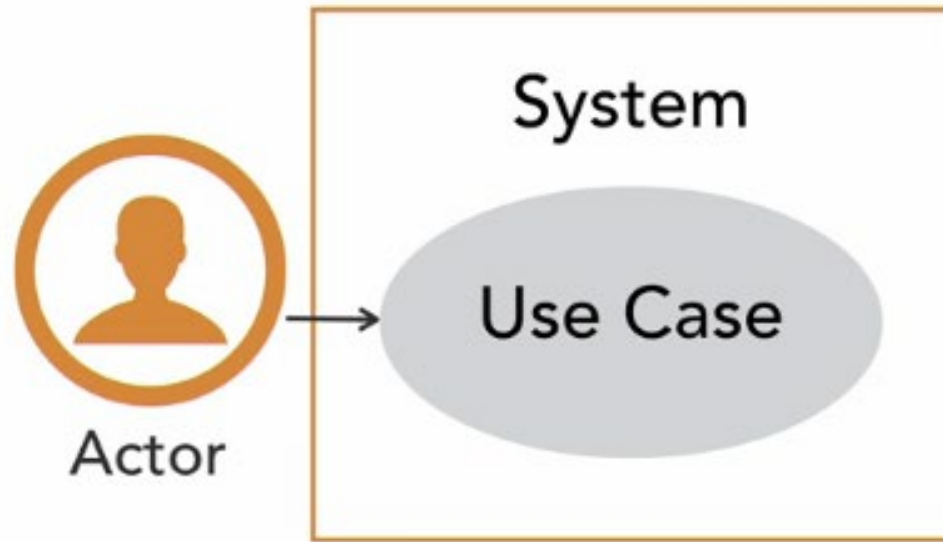
https://en.wikipedia.org/wiki/List_of_Unified_Modeling_Language_tools



Use Case Diagram

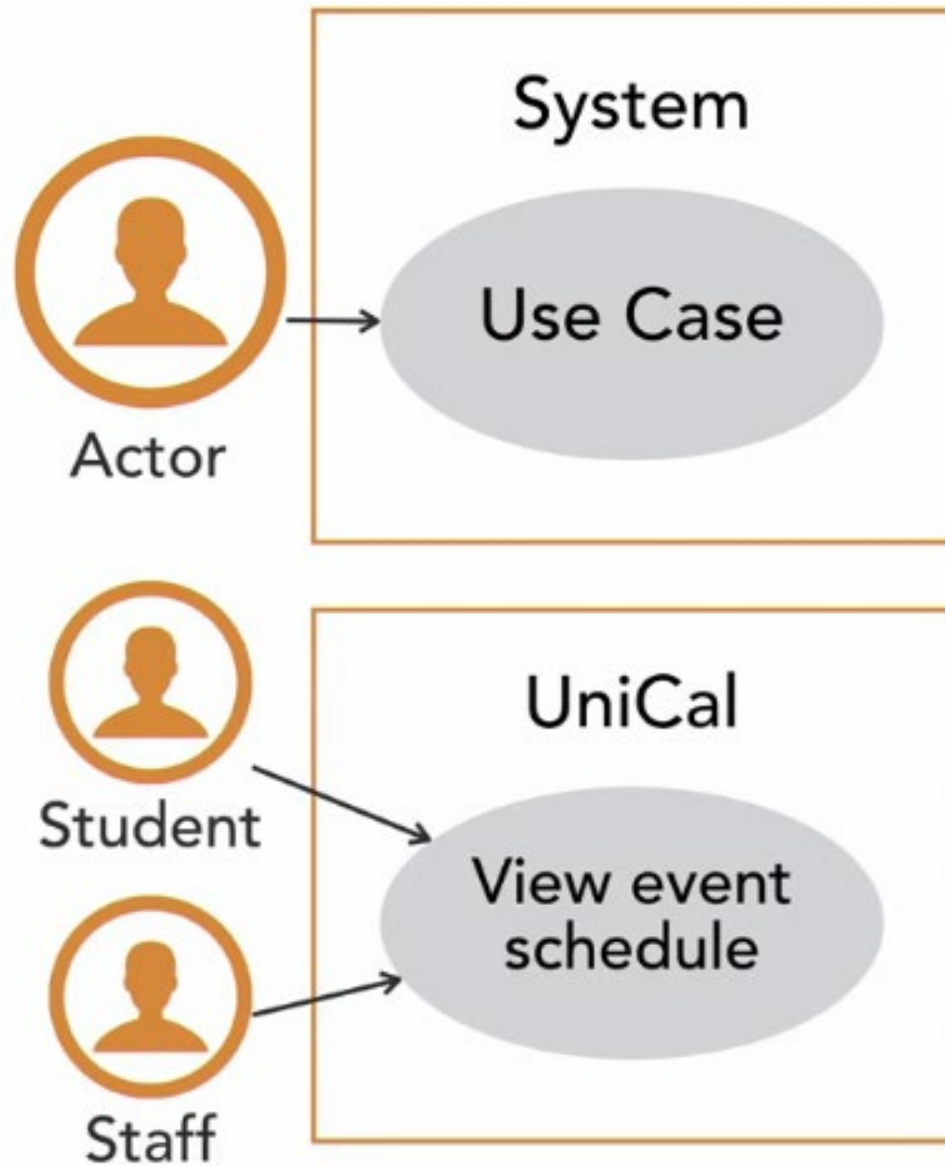
Capture high-level functionality of a system using notations for actors, use cases, and relationships among them

Often drawn by business analysts to depict the summary all use cases in a system



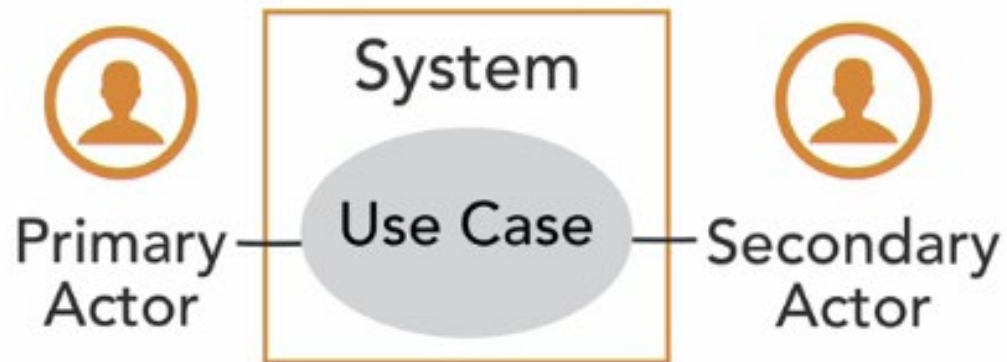
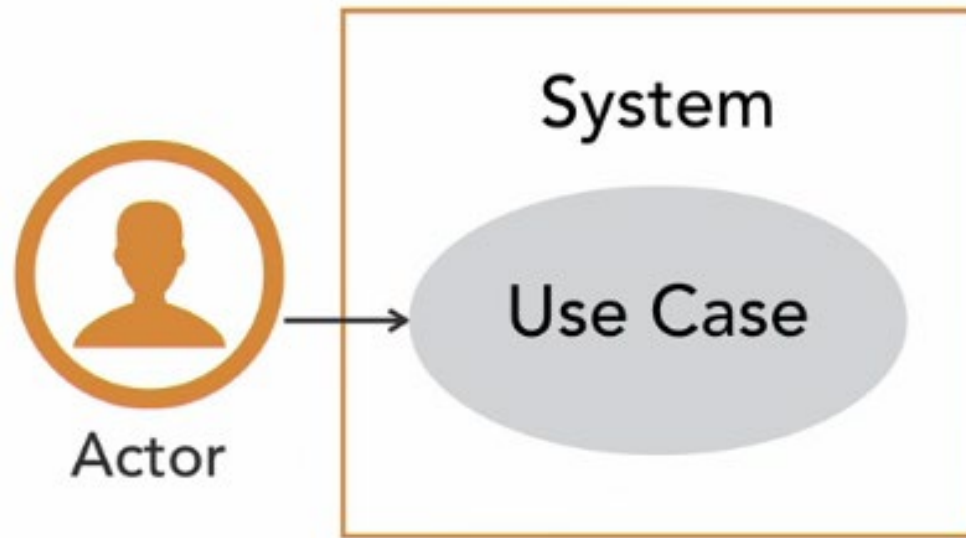
Key Elements

- Use cases
- Systems
- Actors
- Associations



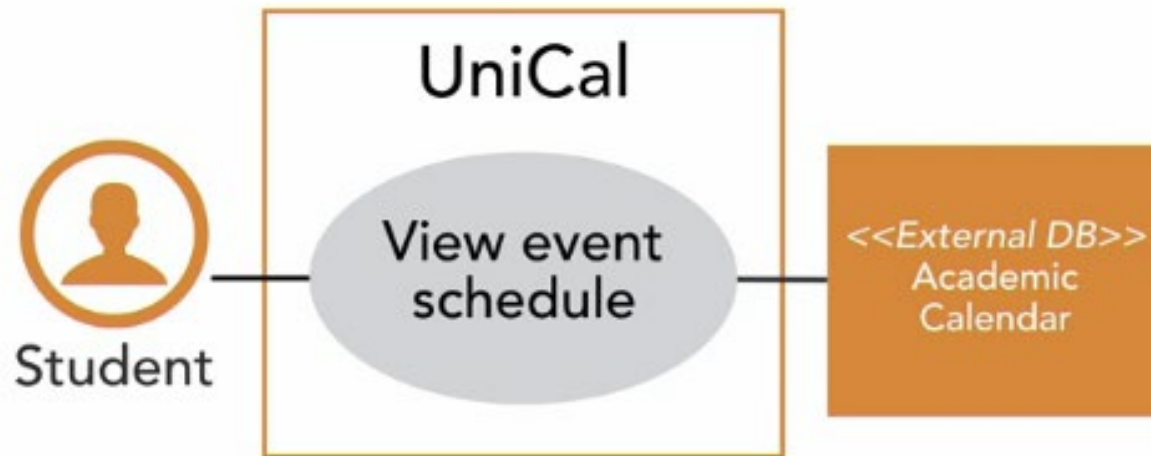
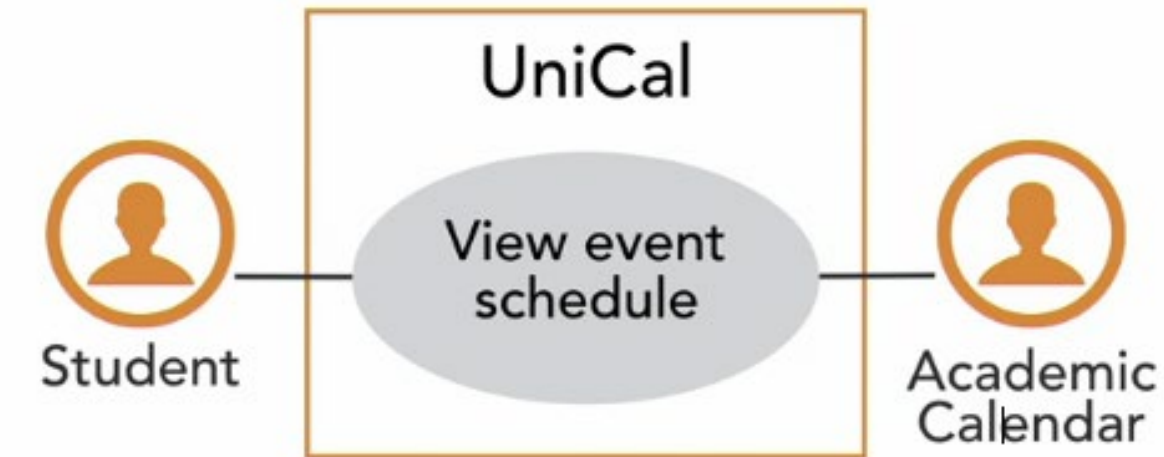
Use Case

- The use case notation is a bubble that carries use case title
- Associated with actors and possibly other user cases



Actor

- A user's role with respect to the system
- May be a human or another system
- Primary actor: whose goal is fulfilled by the use case
- Secondary actor: who is involved in the use case

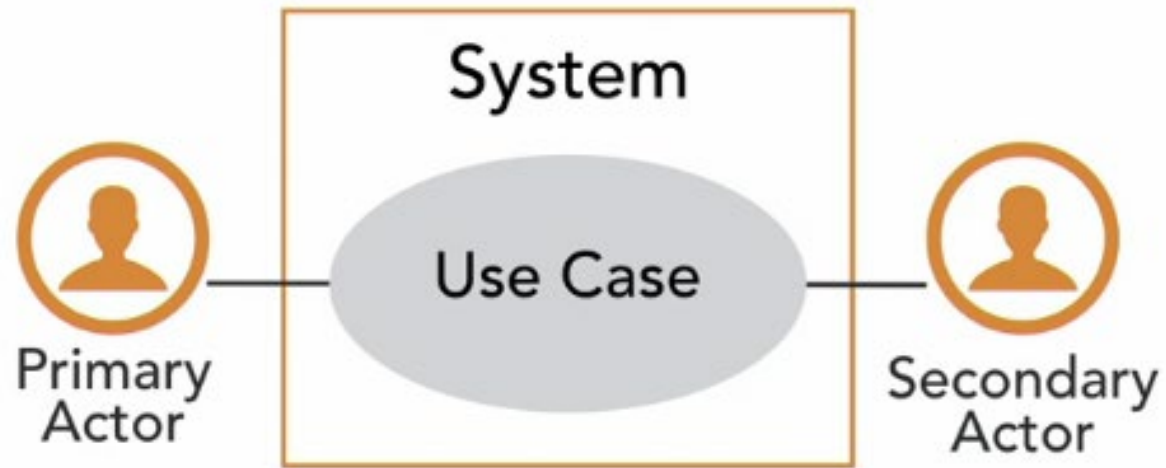


Secondary Actor

Often an external system

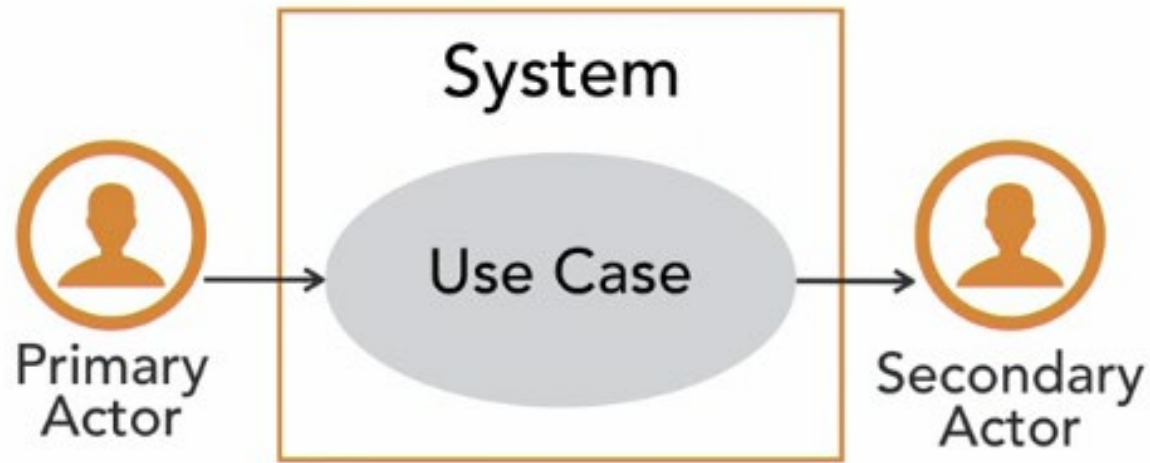
Possible notations

- A nonhuman notation such as a rectangle
- Stereotyped with << >>



Association

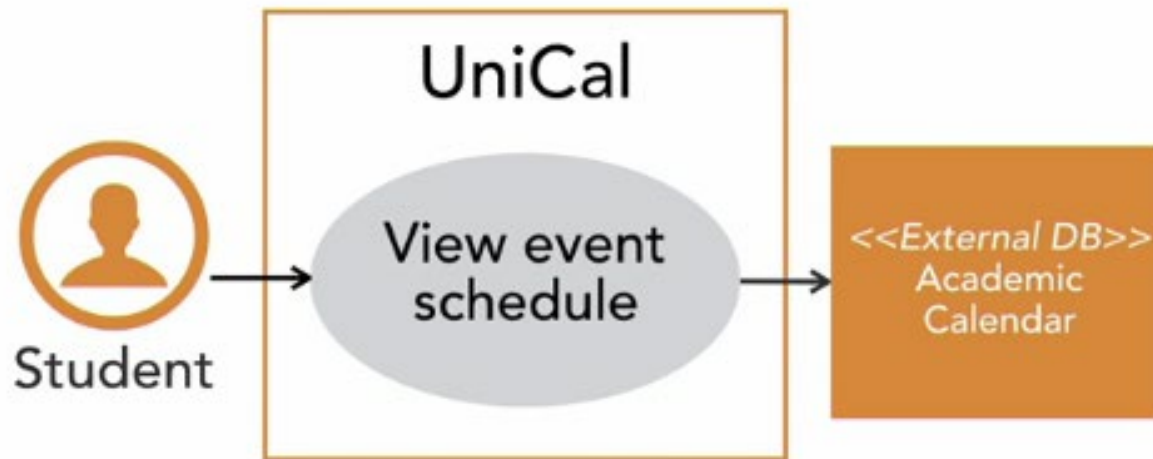
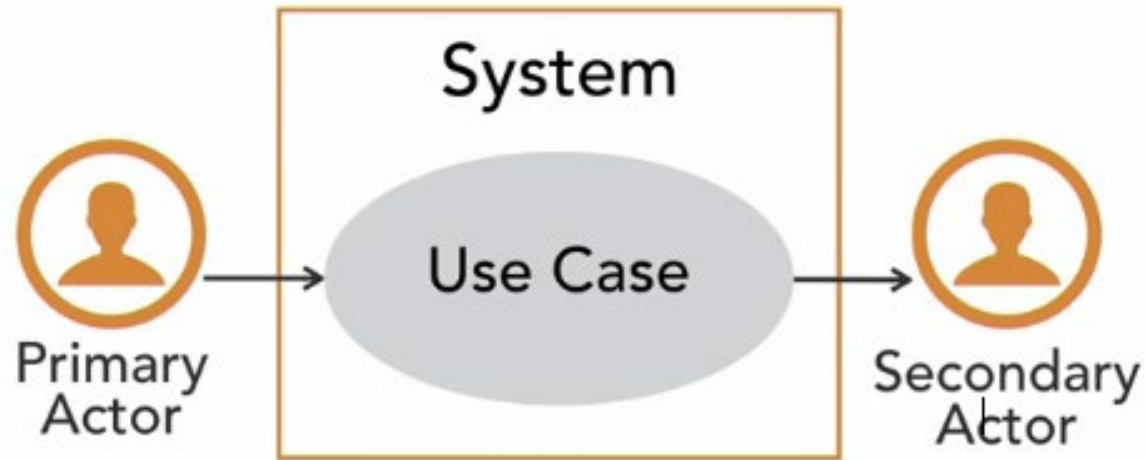
- Between actor and use case
- Between use cases
- Between actors



Between Actor and Use Case

Convention

- Primary actor on the left
- Secondary actor on the right of the system



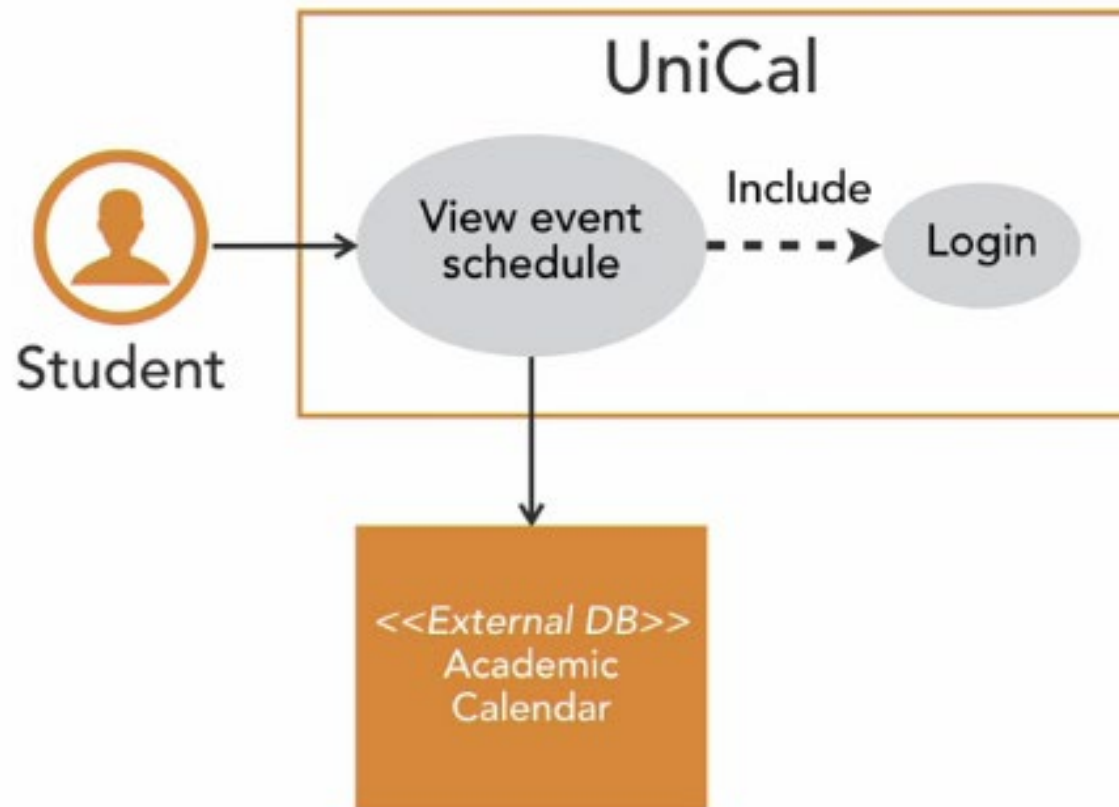
Between Actor and Use Case

Convention

- Primary actor on the left
- Secondary actor on the right of the system

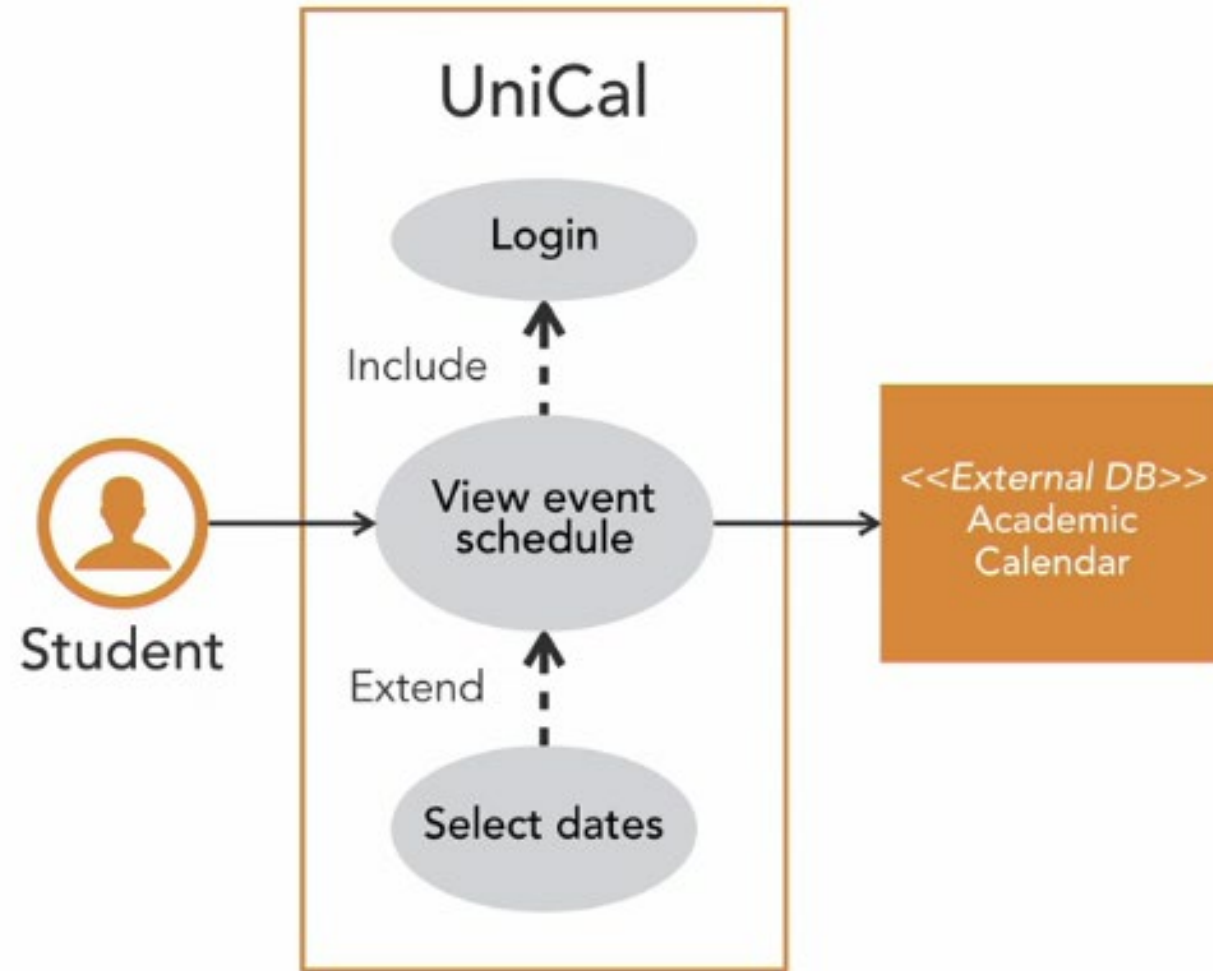
Arrowheads

- From primary to use case
- From use case to secondary



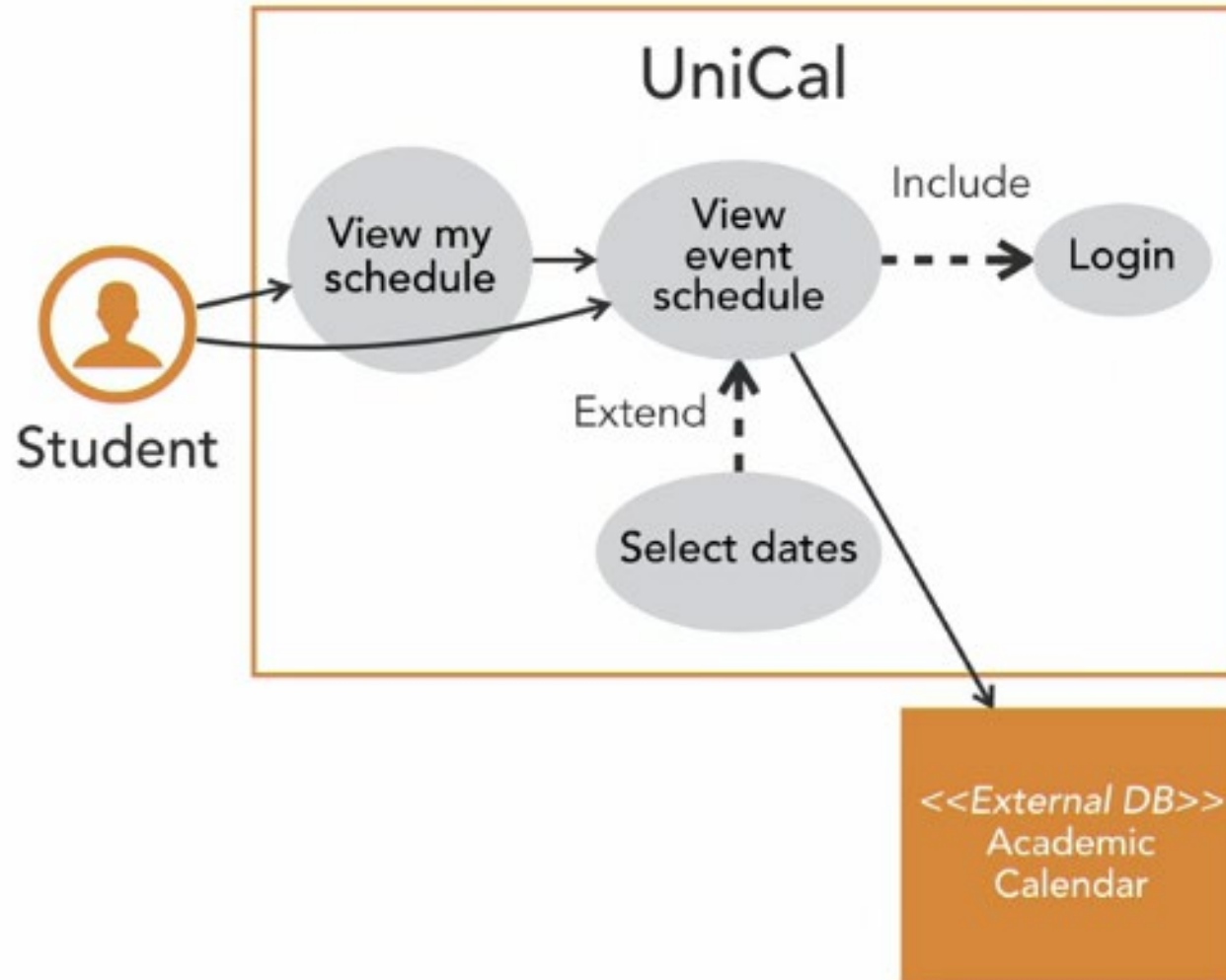
Between Use Cases: Include

- For reusable parts of behavior across two or more use cases



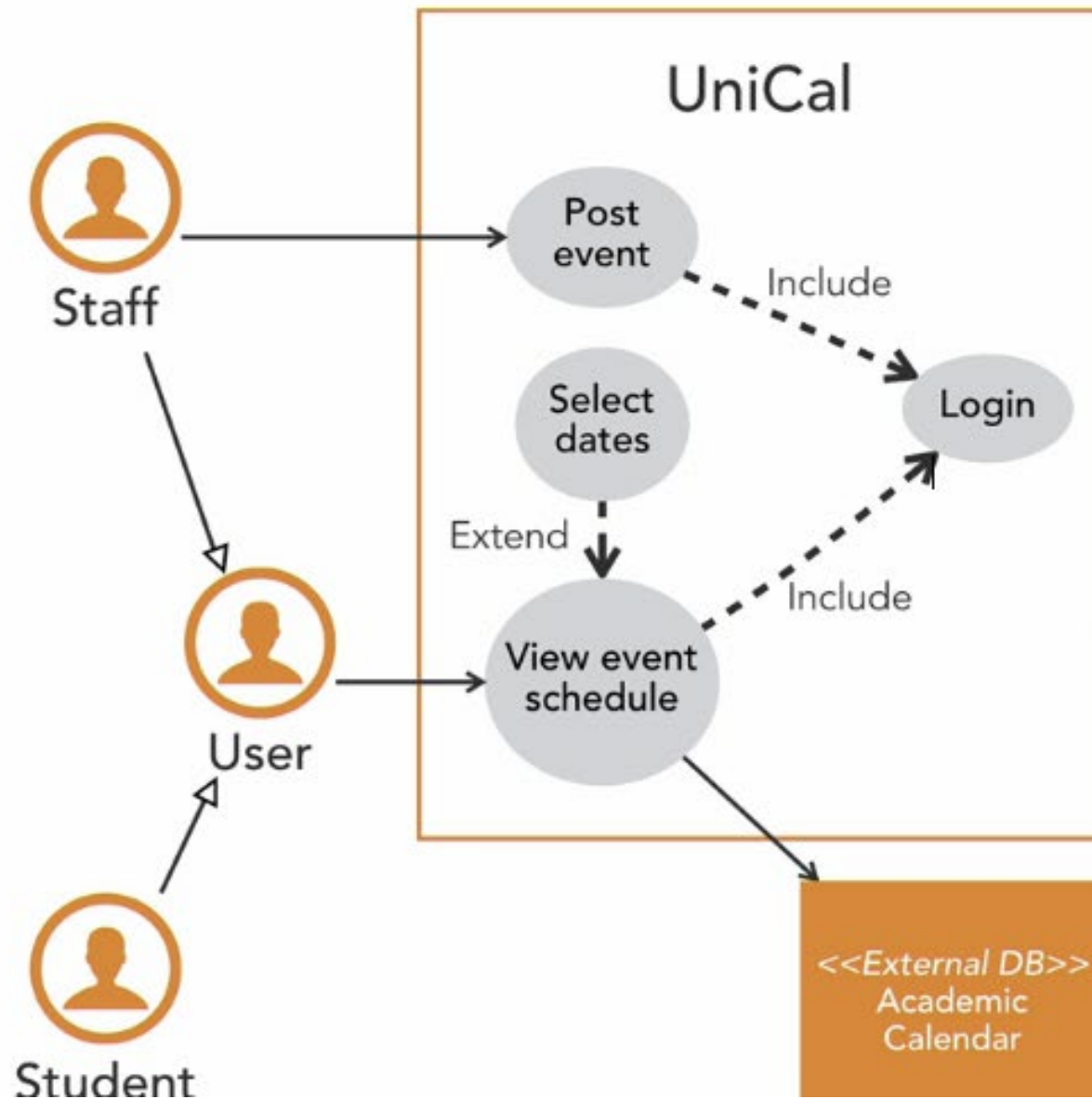
Between Use Cases: Extend

- Optional behavior added to use case
- Helps keep the base use case unchanged while adding more specifics or conditional changes



Between Use Cases: Generalization

- One use case is a special case of another more general use case



Between Actors: Generalization

- Depict generalization – specialization or inheritance relationship between actors