

Introduction to Software Design with Domain-Driven Design

Parinya Ekparinya

Parinya.Ek@kmitl.ac.th

Outline

- ❖ Models
- ❖ Domains
- ❖ Domain-Driven Design
- ❖ Ubiquitous Language

យុទ្ធសាស្ត្រ

ការគោរព

Software Design

- ❖ Software Design is about modelling software **systems**.

ឯកត្រាជាមុនខ្លះ | ចិបចិត្តទាំងនេះ ទាំង ២ នៅក្នុងការរៀបចំសាស្ត្រភាគជាមុន

- ❖ "A system is an organized or complex whole: an assemblage or combination of things or parts forming a complex, unitary whole." (von Bertalanffy, 1935).

- ❖ Parts of a system are known as sub-systems.

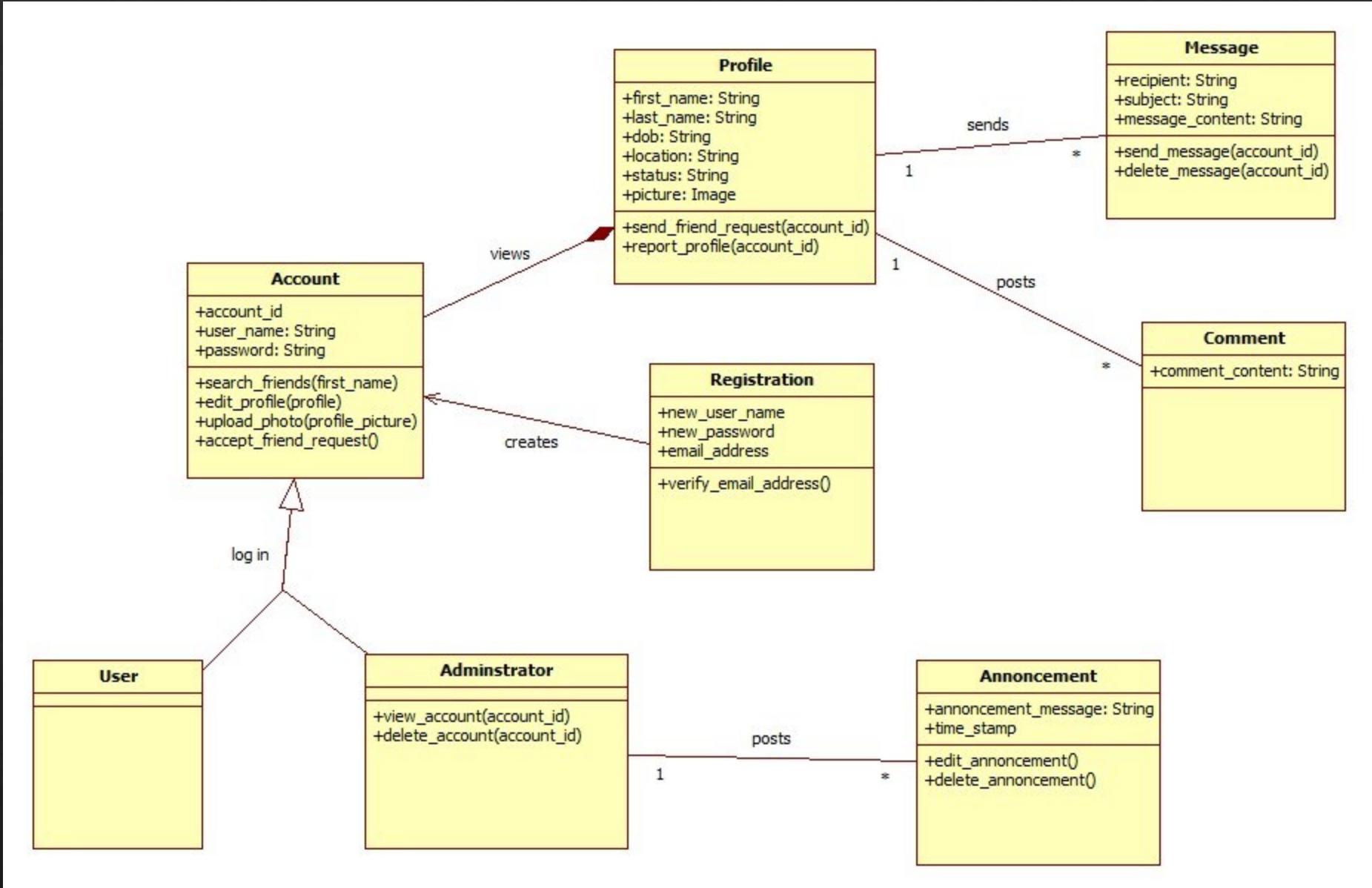
ເກີຍກິບງາມກຳແນ່ງຈຳວົດ

ព្រះរាជាណាចក្រកម្ពុជា

- ❖ Modelling a system means identifying its main characteristics, states and behavior using a notation. 
 - ❖ Unified Modeling Language (UML) is commonly used for modelling software systems.

Unified Modeling Language (UML)

- ❖ The UML is a graphical language for visualizing, specifying, constructing, and documenting the artifacts of a software-intensive system.
- ❖ The UML offers a standard way to write a system's blueprints, including conceptual things such as business processes and system functions as well as concrete things such as programming language statements, database schemas, and reusable software components.
- ❖ The UML was developed at Rational Software in 1994–1995.
- ❖ The UML was also published by the ISO as an approved ISO standard in 2005.
- ❖ The latest version is 2.5.1, which is published in 2017.
For specification: <https://www.omg.org/spec/UML/2.5.1/About-UML/>



Modelling

សម្រាប់

Reason for modelling

- ❖ To understand ឱ្យដឹង
- ❖ To clarify ការឃើញចាន់ចាន់ ដើម្បីចាប់រក្សា
- ❖ To analyze គិតភាព
- ❖ To experiment ពិនិត្យ
- ❖ To evaluate ប្រាក់ប្រាក់

How to model

- ❖ Textual
- ❖ Graphical រាង តាមអ្នស៊ី
- ❖ Mathematical សម្រាប់គិតវិទ្យា

But what is a “Model” ?





“

ក្នុង ...
ជិត្យការអនុការណ៍ដែលបានរាយ domain ទីនេះគឺជាសម្រាប់ការរាយ domain ទាំងអស់

A system of abstractions that describes selected aspects of a domain and can be used to solve problems related to that domain.

”

The definition of model

Evans, E. (2015). Domain-driven design reference. *Definitions and Pattern Summaries*.

But what is a “Domain” ?

កីឡា ? Definition of domain noun from the Oxford Advanced Learner's Dictionary

ជំនួយ / របៀប ការងារ or ព័ត៌មាន

- ◆ an area of knowledge or activity; especially one that somebody is responsible for
 - ◆ Financial matters are her domain. ដីន Topic ជំនួយ ដីនសាធារណៈ (ការចាប់ចិត្តភ័យ)
 - ◆ Physics used to be very much a male domain.
 - ◆ things that happen outside the domain of the home ជំនួយខែគ្រោះ
- ◆ not to be confused with another definition:
(computing) a set of websites on the internet that end with the same group of letters, for example '.com', '.org'

No domain name

“

កំណត់របាយការ

A sphere of knowledge, influence, or activity. The subject area to which the user applies a program is the domain of the software.

”

The definition of domain

Evans, E. (2015). Domain-driven design reference. *Definitions and Pattern Summaries*.

օնակ կ դ ժ ա վ ո ւ գ ա մ ի → Dev գ ա ր ա ն
→ օ ն ա կ ս մ ա լ ա կ ա ն ա կ ա ն ա կ ա ն

DOMAIN-DRIVEN DESIGN (DDD)

A software design approach focusing on modelling software to match a domain according to input from that domain's experts.

մ ա

մ ա ք ա ս

պ ա շ ա ն ա ն ա ն

ո հ ո շ ա յ ա ն ա ն ա ն

Software Domains

- ❖ Every software program relates to some activity or interest of its user.
- ❖ That subject area to which the user applies the program is the domain of the software.
- ❖ Some domains involve the physical world:
The domain of an airline-booking program involves real people getting on real aircraft.
- ❖ Some domains are intangible:
The domain of an accounting program is money and finance.
- ❖ Software domains usually have little to do with computers, though there are exceptions:
The domain of a source-code control system is software development itself.

Evans, E., & Evans, E. J. (2004). *Domain-driven design: tackling complexity in the heart of software*. Addison-Wesley Professional.

Why Domain-Driven?

- ❖ Software is made up of code. កម្រិត
ក្រឡាក់នៃកម្រិត
- ❖ Seeing the software as simply objects and methods, some developers might spend too much time solely on code.
ការរក្សាទុក កំណត់វា កែប្រែទីនេះ
- ❖ To create good software, however, one must know what that software is all about.
ការរក្សាទុក កំណត់នៃកម្រិត
- ❖ The entire purpose of the software is to enhance a specific domain. To be able to do that, the software has to fit harmoniously with the domain it has been created for.
ការរក្សាទុក កំណត់នៃកម្រិត
- ❖ Software needs to incorporate the core concepts and elements of the domain, and to precisely realize the relationships between them. 

Domain Knowledge

- ❖ To create a good banking software system, one must understand the domain of banking.
- ❖ Who knows banking?
 - ❖ The software analyst?
 - ❖ The software architect?
 - ❖ The developer?
 - ❖ The bankers? *Domain experts*
- ❖ Models are tools for dealing with complexity of information required.

“A domain model is not a particular diagram; it is the idea that the diagram is intended to convey. It is not just the knowledge in a domain expert’s head; *it is a rigorously organized and selective abstraction of that knowledge.*”

DML คือ เอ็นด์ฟอร์ก้า Class Diagram
ที่ใช้สีน้ำเงิน Class Diagram

Evans, E., & Evans, E. J. (2004). *Domain-driven design: tackling complexity in the heart of software.* Addison-Wesley Professional.

ការសិក្សា D. experts នៃបច្ចេកវិទ្យា

Communicate with Domain Experts

- ❖ Event storming ឯកសារ work shop
 - ❖ A workshop-based method to quickly find out what is happening in the domain of a software program.
 - ❖ The result is expressed in sticky notes on a wide wall.
- ❖ Domain model diagram
 - ❖ In UML, a class diagram is used to represent the domain model.

Event
storming



By Henning Schwentner - Own work, CC BY-SA 4.0, <https://commons.wikimedia.org/w/index.php?curid=57766348>

ស្រីលីអិន ធម្ម. ០៩៨

The Utility of a Model in Domain-Driven Design

ឈានបរាំ២០១៧

កំណត់ការអង្គភាព

1. The model and the heart of the design shape each other.

ឈាន ខ្លួន

រាជធានីភ្នំពេញ ទី ៣ សម្រាប់ សាស្ត្រ និង វិទ្យាឌភាព

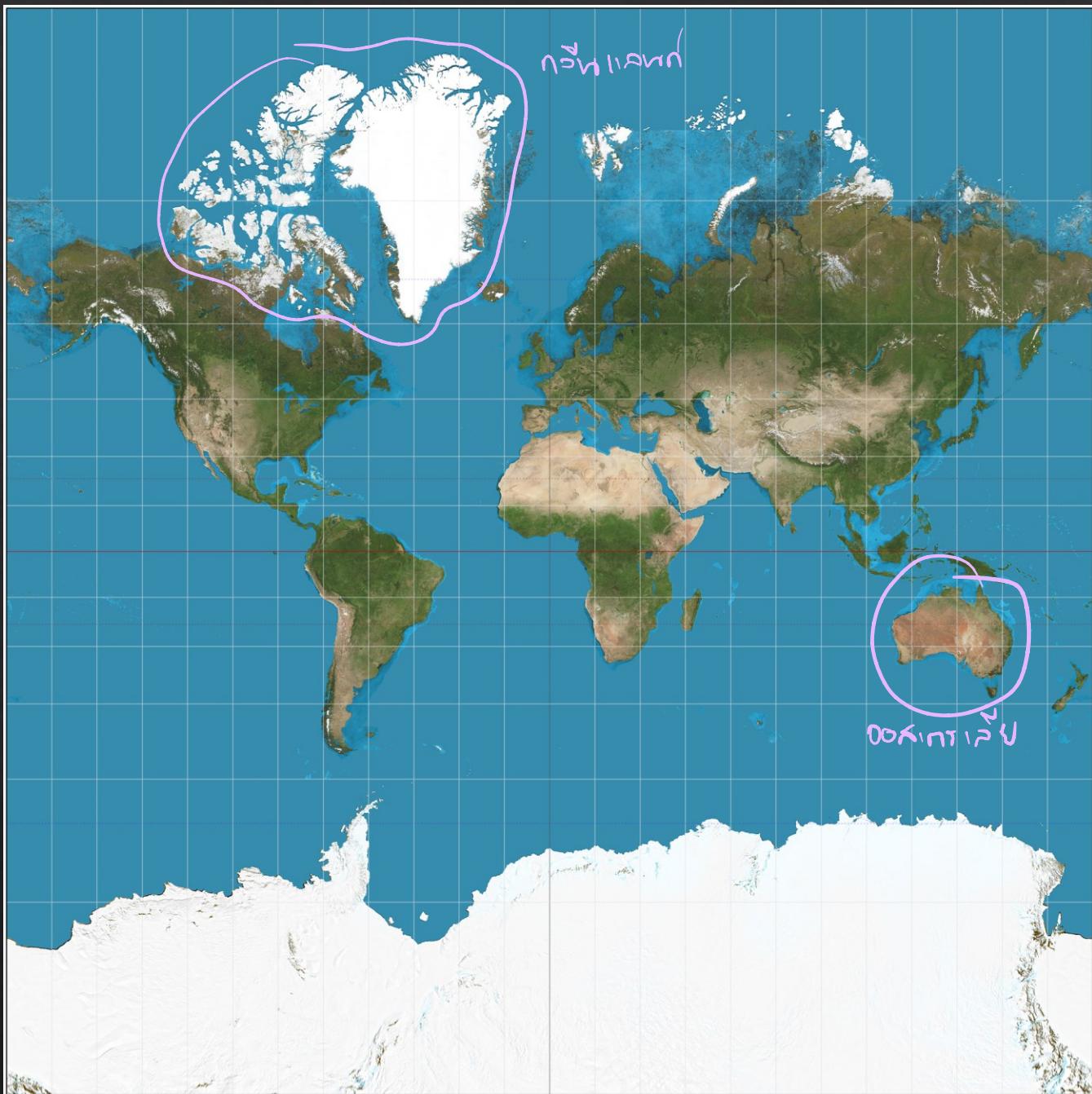
2. The model is the backbone of a language used by all team members.

ស៊ី គុណភាព សកម្មិយោ ថែរិនការអនុវត្តន៍

3. The model is distilled knowledge.

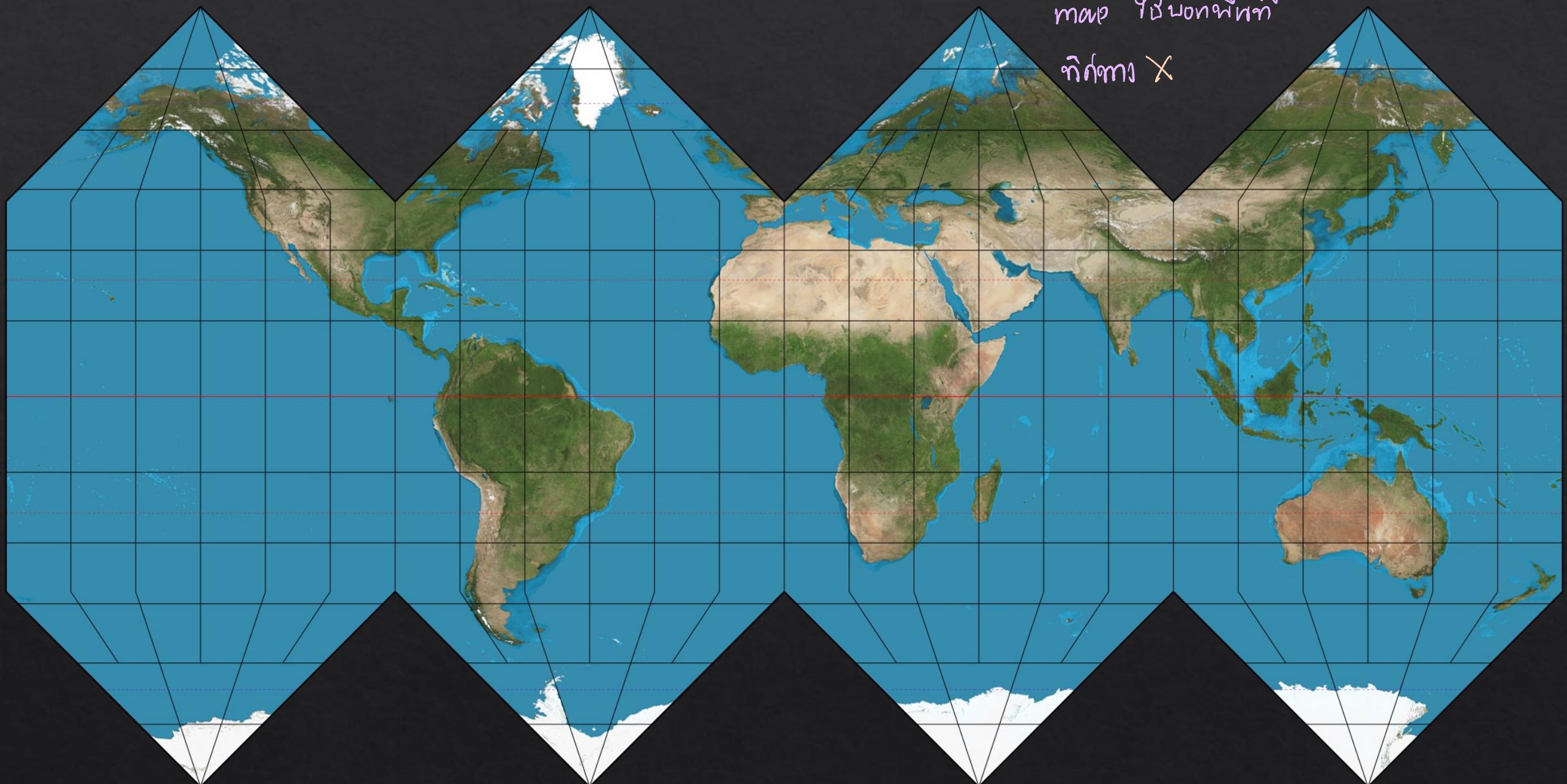
Evans, E., & Evans, E. J. (2004). *Domain-driven design: tackling complexity in the heart of software*. Addison-Wesley Professional.

Scenario 1: World Maps



map นี้ 95% ก้อนหิมะ
อีสาน กำเนิดในภูมิภาค

map စုပေါင်မျဉ်
ကိုယ် X



Questions for Scenario 1

- ❖ Please look at the maps from previous slides and compare the size of Australia and Greenland on both maps.

ກົມົງກວ່າ 3.6 ນາມ

- ❖ Whose area is larger, Australia or Greenland?

- ❖ Why these maps present information differently? ກໍາລູມເວລີ ອຸປະນຸມຕົກ ? ດາວອຸດປະກາດ

- ❖ Which map is right? Why?

↳ model ທີ່ມາດຕະກຳທີ່ຕ້ອງການ = ສອງ ປົກສົງ

Scenario 2: Airplane Flight Control System

Page 8 of Domain Driven Design Quickly by [InfoQ](#)

Fern

The Ubiquitous Language

ຊູຕກິນີ້ຈຳໃດຕາງກົດ (ການາກລາວ)

The Need for a Common Language

(Dev)

កំពង់អាមេរិក

- ◆ DDD suggests that the software specialists must work with the domain experts to develop models. Such an approach usually has some difficulties due to a communication barrier.
 - ◆ The developers are often familiar with classes, methods, and algorithms. They want to know what classes to create and relationships among them.
 - ◆ The domain experts usually know nothing about software libraries, frameworks, and in many case not even databases. They know about their specific area of expertise.
 - ◆ To overcome the communication difficulties, software specialists and domain experts need to speak the same language when they meet to talk about the model and to define it. *බෙජපුද්ගලීකාණ්ඩ රූප*
 - ◆ Make sure this language appears consistently in all the communication forms used by the team; for this reason, the language is called the **Ubiquitous Language**.

ជាករណីការបង្ហាញពីរាជរដ្ឋ
ជាករណីរោងចាយសំខាន់

ទ. សរុប្បែ ??

Creating the Ubiquitous Language

ស្នើសុំអង់គេយ

- ❖ Languages do not appear overnight. It takes hard work and a lot of focus.

លាតីដី ការងារ ទូទាត់ និងស្នើសុំ / ហាក់ចិត្តការណ៍ ទាញរាយ និងស្នើសុំ = ភូមិការ

- ❖ Experimenting with alternative expressions, which reflect alternative models. រួមចិត្តការណ៍ ក្នុងការរាយការ

- ❖ Then refactor the code, renaming classes, methods, and modules to conform to the new model.

វិអីវិវាទ = ចាយជីវិត ធម៌រាយការ

- ❖ If domain experts cannot understand something in the model or the language, then it is most likely that there is something is wrong with it.

Scenario 2 continue: Airplane Flight Control System

Page 16 of Domain Driven Design Quickly by InfoQ

But what exactly is a “Ubiquitous
Language” ?

“

A language structured around the domain model
and used by all team members
within a bounded context to connect
all the activities of the team with the software.

”

The definition of ubiquitous language

Evans, E. (2015). Domain-driven design reference. *Definitions and Pattern Summaries*.

မြန်မာ

But what is a “Context” ?

“

The setting in which a word or statement appears
that determines its meaning. *Statements
about a model can only be understood in a context.*

Ex អ្នករាយកំពង់នៅខែមីនាំទី០១ នឹងបានកំណត់ថ្មីៗ

”

The definition of context

Evans, E. (2015). Domain-driven design reference. *Definitions and Pattern Summaries.*

ករណីសម្រាប់អង្គភាព

How about “Bounded Context” !?

It is a bit complicated to explain the term here. So, let us keep it for some time later.
For the time being, you can think of a bounded context as a boundary of a domain.

Summary

Domain-Driven Design is an approach to the development of complex software in which we:

1. Focus on the core **domain**. **Focus**
2. Explore **models** in a creative collaboration of domain practitioners and software practitioners.
3. Speak a **ubiquitous language** within an explicitly **bounded context**.