

Multi-Platform Chatbot Modeling and Deployment with the Xatkit Framework

gdaniel@uoc.edu
jordi.cabot@icrea.cat

Once upon a time



Let's create a chatbot to help newcomers to write issues on Github!

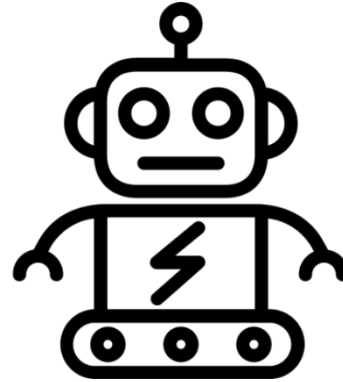
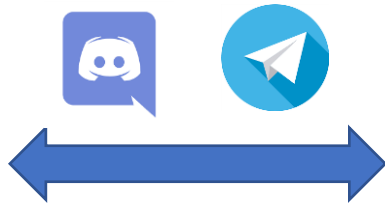
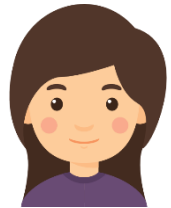
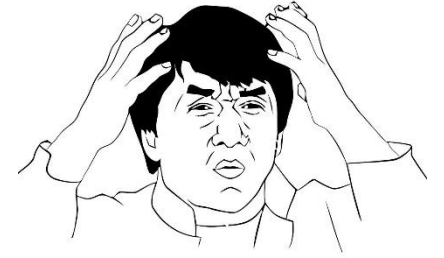
Alright! It's just a set of questions & answers, this will be pretty simple! 🌸



Narrator

It wasn't.

What went wrong?



```
if(" I have an issue") {  
  return "In which repository?"  
}
```

Chatbots are complex systems

Text Processing

Deployment

Conversation Logic

Evolution

Maintenance

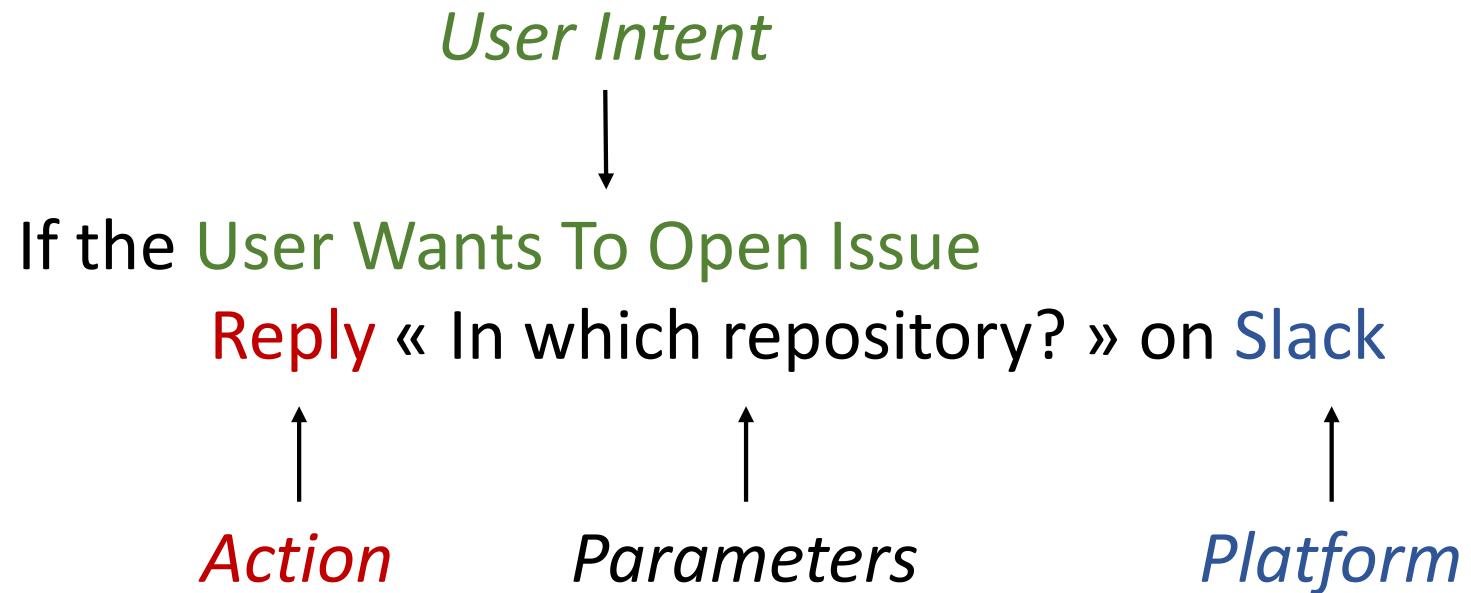
Messaging
Platforms

External Services

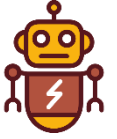
Tests

Chatbot applications

In the end what do we **really** want to do?

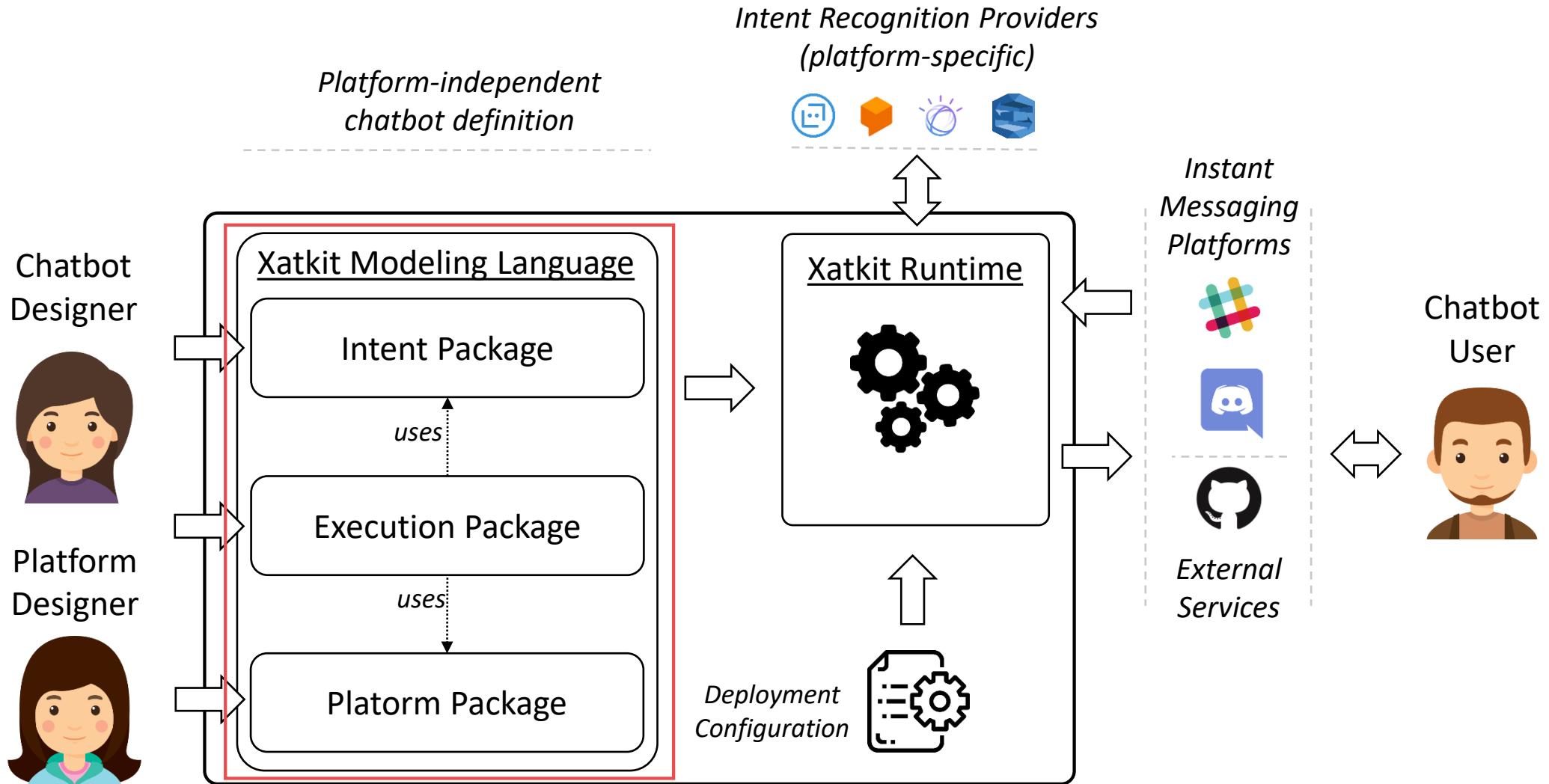
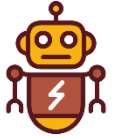


Xatkit Framework

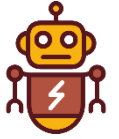


- Raise the level of abstraction at what chatbots are defined
 - Focus on the core logic of the chatbot
 - Conversation and user interactions
 - Action computations
 - Independent from specific implementation technologies
- Automate the deployment and execution of the modeled chatbot
 - Deploy over multiple platforms
 - Execute the chatbot logic

Xatkit Framework



Xatkit Framework



- Execution language package

Import library Example

Import platform Slack

Import platform Discord

Import platform Github

Listen to Slack

on intent OpenNewIssue **do**

```
Slack.Reply("Sure, I'll help you to write  
your issue! Which repository would you  
like to report an issue for?")
```

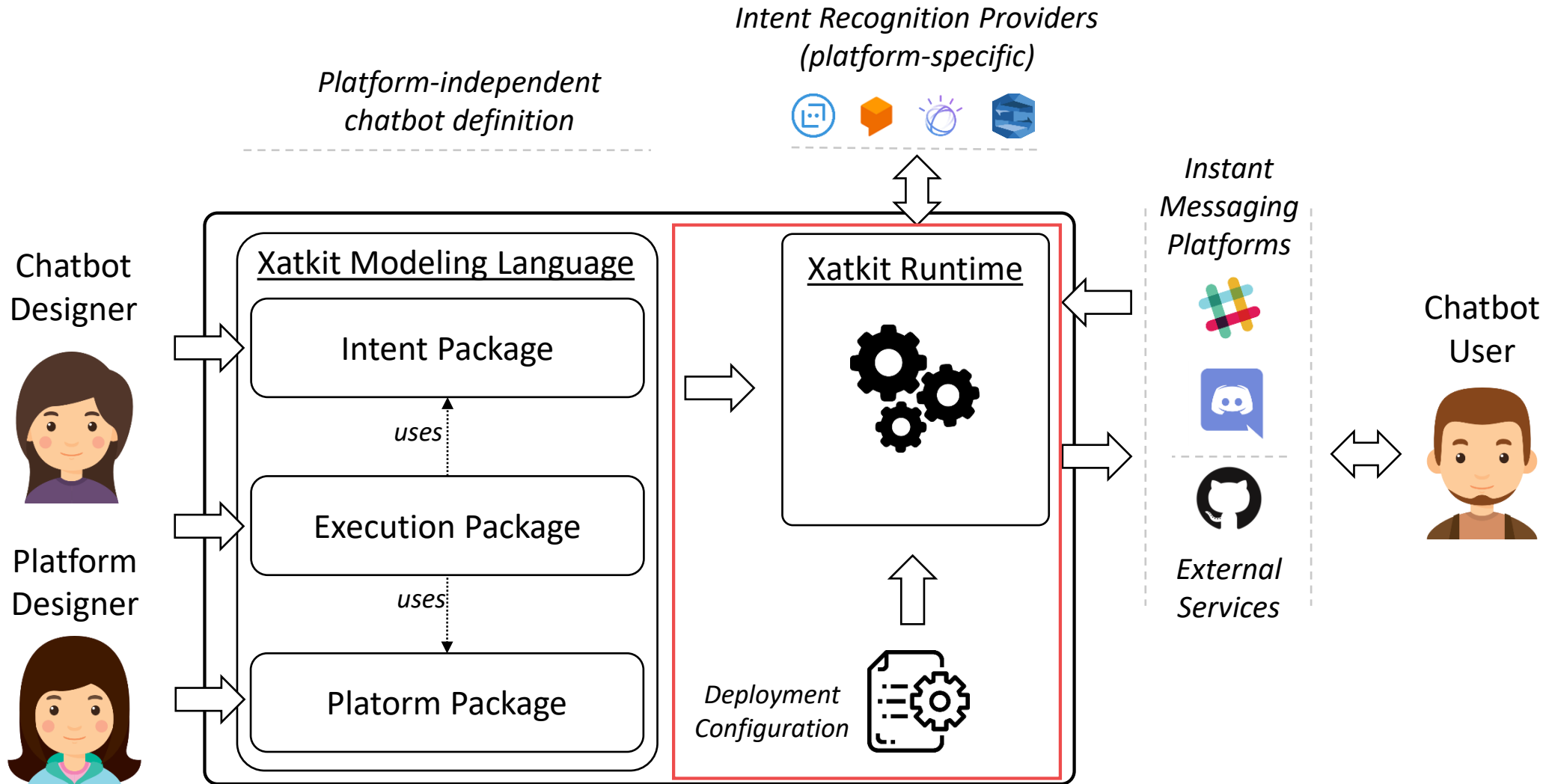
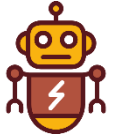
on intent SpecifyRepository **do**

```
Slack.Reply("Alright, your issue is related to  
repository" + context(Repository).get(name))  
[...]
```

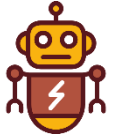
```
Github.OpenIssue(context(Repository).get(name),  
context(Issue).get(title),  
context(Issue).get(content))
```

```
Discord.PostMessage("A new issue has been opened in "  
context(Repository).get(name))
```


Xatkit Framework

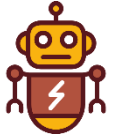


Xatkit Framework



- Runtime component
 - Generic event-based execution engine
 - Platform-specific connectors
 - Slack, Discord, Web-based, Giphy, Github, ... (<https://github.com/xatkit-bot-platform/xatkit-releases/wiki>)
 - Automatic deployment
 - Execution life-cycle
 - Inputs
 - Chatbot model (defined with the Xatkit modeling language)
 - Configuration file

Applications – Theme 100% reusable



- Digital transformation for public administration
 - Help locate and access online resources
 - Integrate city Open Data (pollution API, carbon footprint, etc)
 - Any bot reducing public administration paper consumption
- Educational chatbots (advanced Q&A)
 - Recycling practices
 - Minimizing wastes
 - Energy consumption
- Ideas are welcome!
 - gdaniel@uoc.edu
 - jordi.cabot@icrea.cat