

Applied Nlp / Nlu

Mihai (Mikey) Uricaru

@muricaru

muricaru@gmail.com

- ❖ 20+ Years in IT
- ❖ Solution Architect
- ❖ 5 years in AI
- ❖ Trying to be platform agnostic

AGENDA

3



Intro to NLP



Inspiration



Examples

A BIT OF HISTORY

4

The first efforts were connected to translation machines

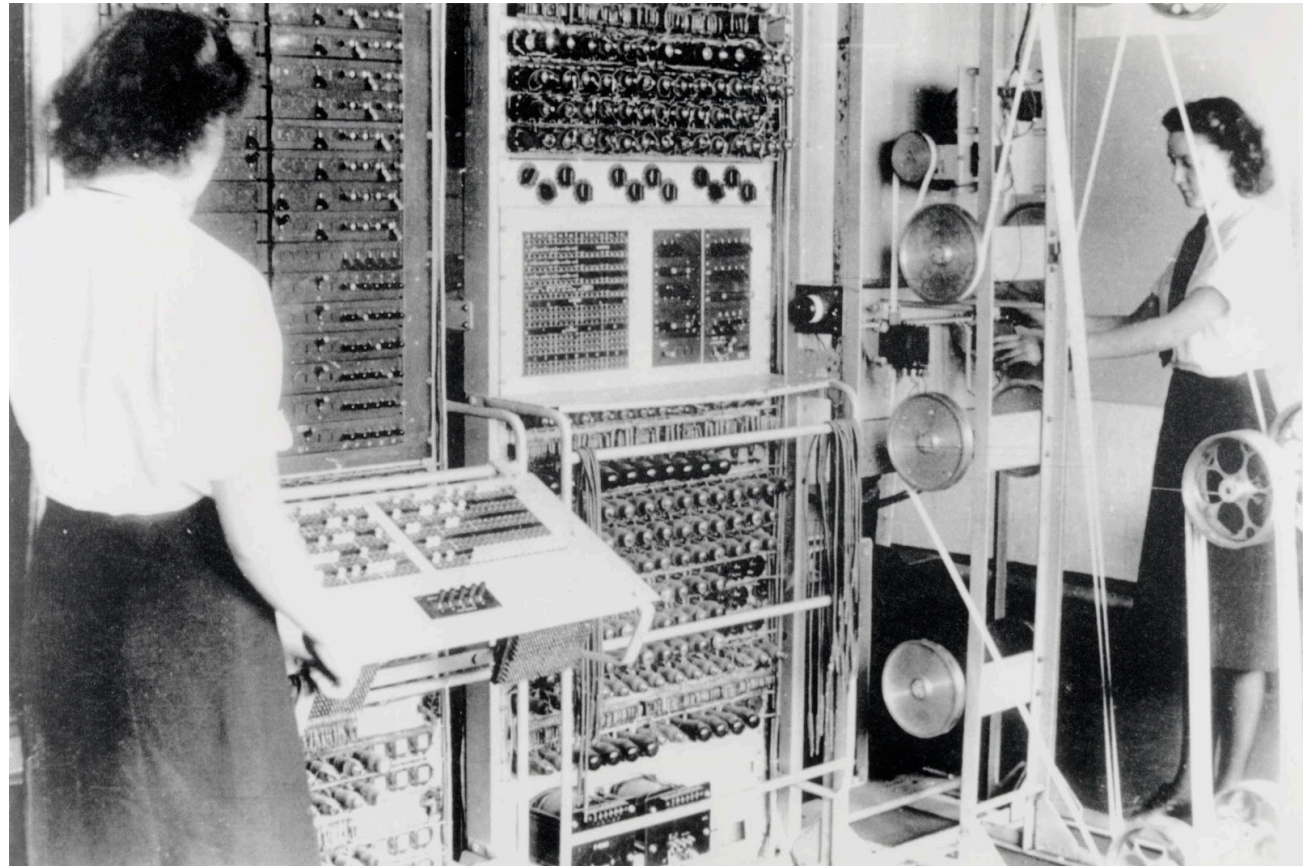
This was the early 30's, so not much was really achieved



A BIT OF HISTORY

5

New era



TURING

6

- ❖ Based on his work during the WWII on deciphering the German encryption, Turing start focusing on the concepts of AI and Natural Language Processing
- ❖ He did ask a fundamental question: “Can machine think ?”
- ❖ The Turing test
- ❖ Loebner Prize

FIRST INDUSTRIAL

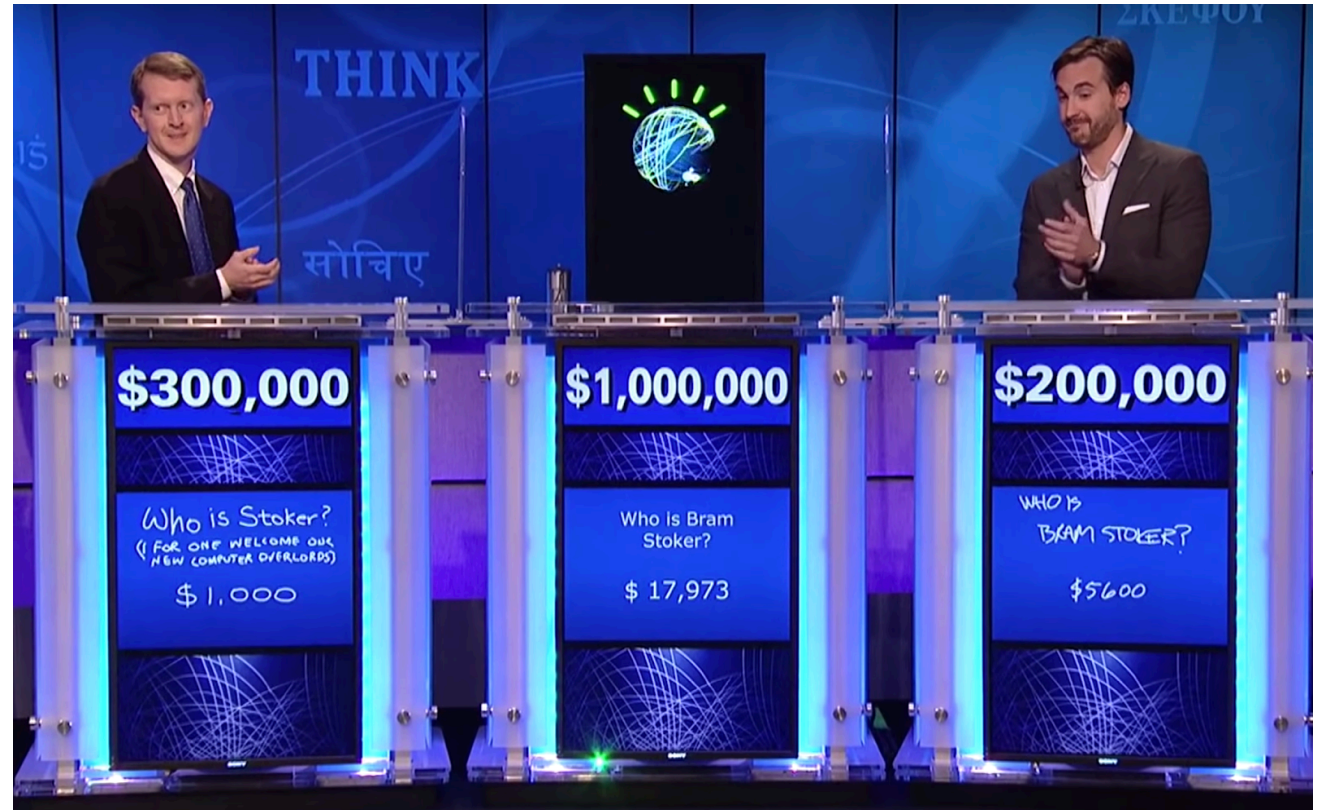
7

- ❖ Many attempts between 50's and 2000 with summers and winters
- ❖ IBM came up with Watson in 2006 - they started the new age, an almost permanent summer
- ❖ Watson went to play Jeopardy! in 2011 and won.

THE CHAMP

8

William Wilkinson's 'An Account of the Principalities of Wallachia and Moldavia' inspired this author's most famous novel.



A BUNCH OF ACRONYMS

9

NLP - Natural Language Processing

NLU - Natural Language Understanding

NLG - Natural Language Generation

V2T - Voice to Text

NLP is a cross section of CS, AI, Data mining and a lot more

WHERE CAN YOU USE IT

10

- ❖ Chat Bots
- ❖ Text processing
 - extracting info / convert unstructured 2 structured
 - classification
 - translation
- ❖ Assistant (Siri, Alexa, Google assistant, Bixby etc)

LINGO 1

11

Entities - Named Entity

Mikey went to **Bucharest** to attend a conference

Intents

Mikey went to Bucharest **to attend** a conference

Coreference

Mikey went to *Bucharest* to attend a conference. **He** had a great time *there*.

LINGO 2

12

- ❖ Sentiment
- ❖ Tone / Emotion
- ❖ Semantic roles
- ❖ Slots - quite specific to Chat Bots

PLATFORMS

13

Cloud

- + easy to start
- + quick tools to test
- + advance features out of the box
- + highly efficient

± some have custom models

- stuck? tough luck
- cost might spiral
- some clients are stuck with on prem

On premise

- + works on edge nodes
- + you have total control
- + strong community
- + data is private

- you will lose a lot of time
- might not be cheaper
- you focus on tweaks not on results
- steep learning curve

COMMERCIAL

14



IBM

- ❖ Natural Language Understanding
 - ❖ Knowledge studio - custom models
- IBM is the veteran of the NLP/NLU market



Amazon Comprehend

AMAZON

Amazon Comprehend - new to market but with great ambitions



GOOGLE

Cloud Natural Language - Some of the features are truly unique



MICROSOFT

Text Analytics API
LUIS

OPEN SOURCE

15

- ✦ **Rasa**
- ✦ Spacy
- ✦ Mitie
- ✦ Stanford NLP
- ✦ NLTK
- ✦ Gensim
- ✦ TensorFlow
- ✦ PyTorch
- ✦ AllenNLP
- ✦ PyText

CHOOSE A PATH

Collection of stories

See how to build Johnny the
virtual devOps

3 quick comparisons

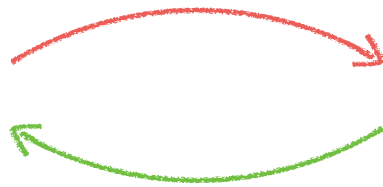
SOME EXAMPLES

17

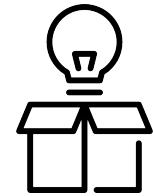
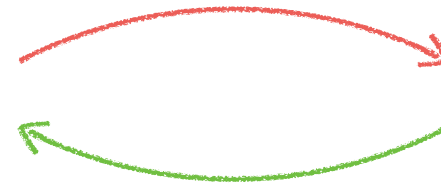
- ❖ Canadian Police
- ❖ Market Research
- ❖ Car diagnostic
- ❖ Customer support satisfaction
- ❖ DevOps assistant
- ❖ Recommendation system - listener

BASIC STRUCTURE

18



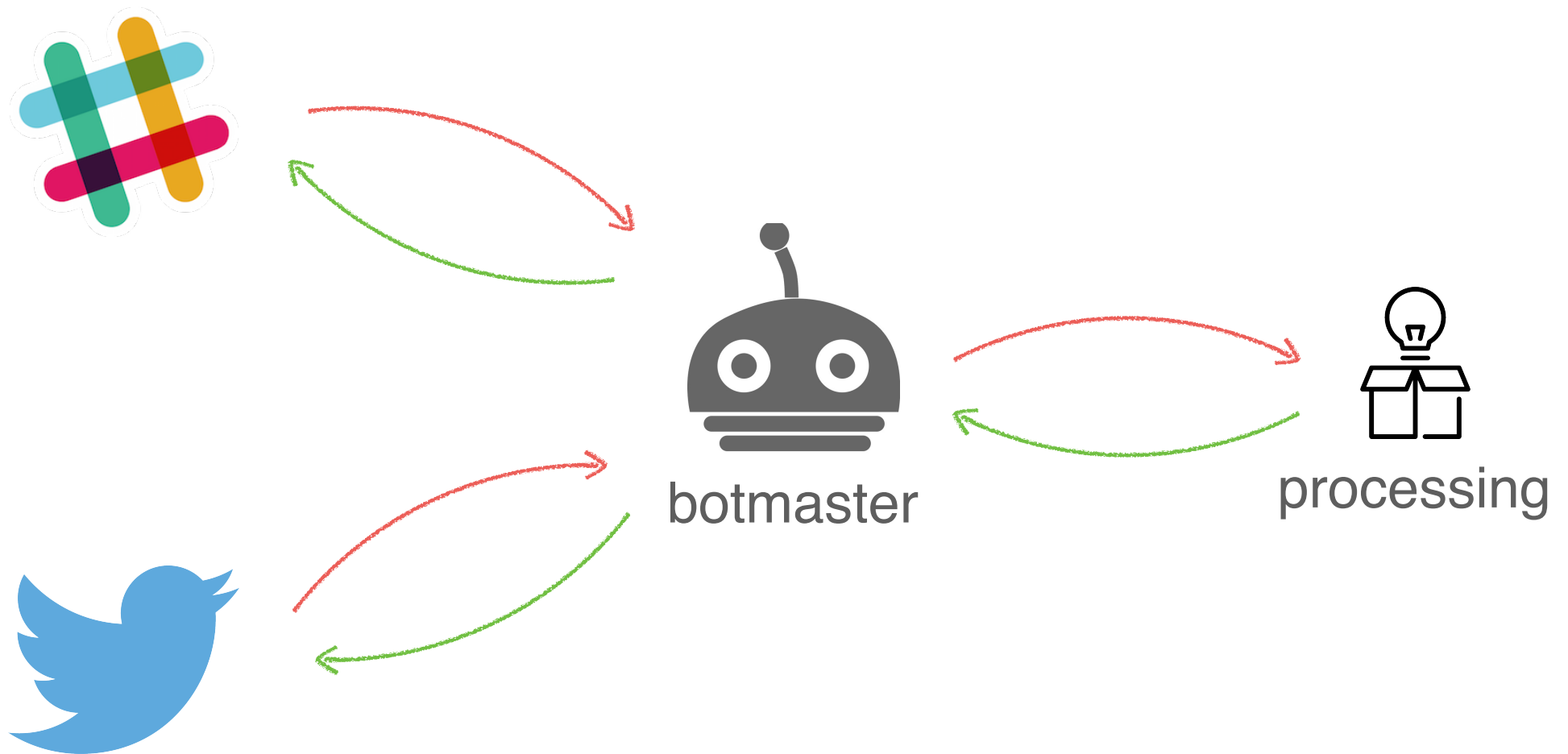
conversational
interface



processing

BOT FRAMEWORKS

19



BOTMASTER

20

```
const Botmaster = require('botmaster');  
const MessengerBot = require('botmaster-messenger');  
const SlackBot = require('botmaster-slack');  
const SocketIoBot = require('botmaster-socket.io');  
const TwitterBot = require('botmaster-twitter-dm');  
const TelegramBot = require('botmaster-telegram');  
  
const config = require('./config'); // your config file with your credentials etc  
  
const botmaster = new Botmaster();
```


BOTMASTER

21

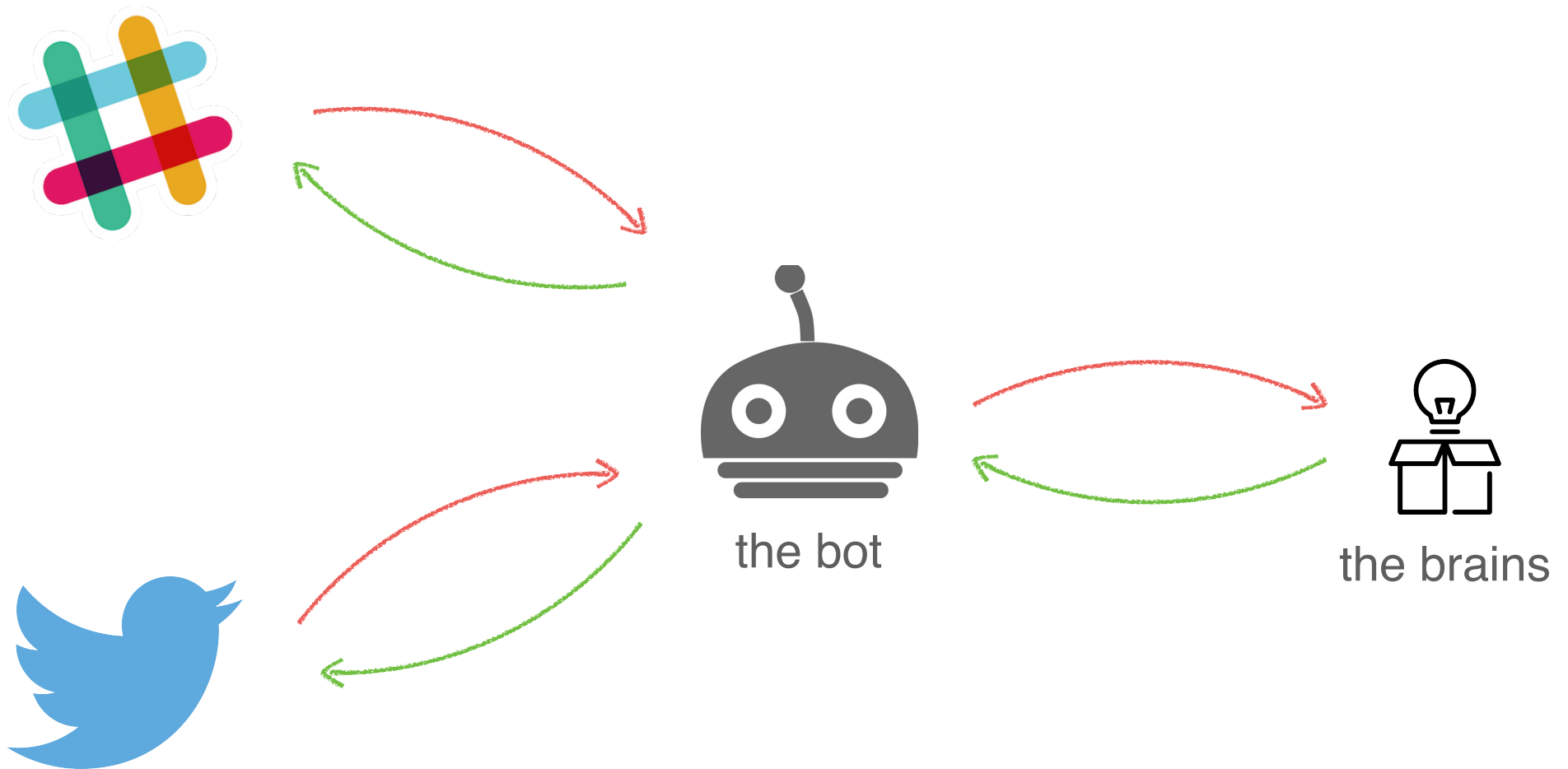
```
const botmaster = new Botmaster();

botmaster.addBot(new MessengerBot(config.messengerSettings));
botmaster.addBot(new SlackBot(config.slackSettings));
botmaster.addBot(new SocketioBot({
  id: 'mySocket.ioId', // whatever you want it to be for socket.io bots
}));

botmaster.addBot(new TwitterBot(config.twitterSettings));
botmaster.addBot(new TelegramBot(config.telegramSettings));
```

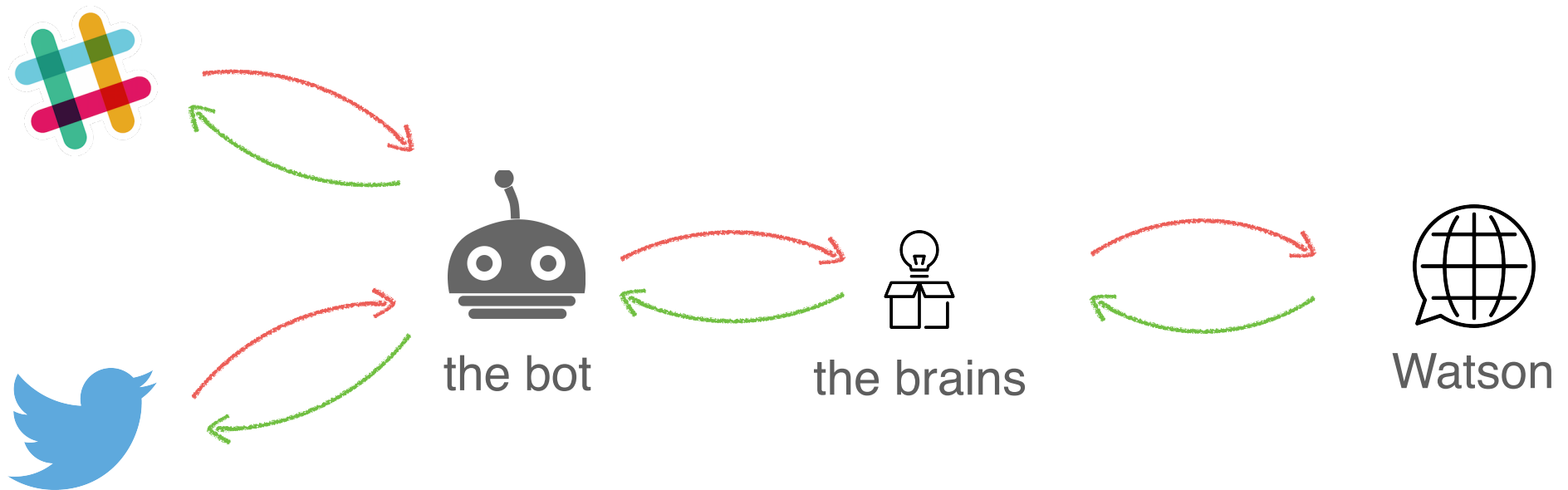
UNDERSTANDING

22



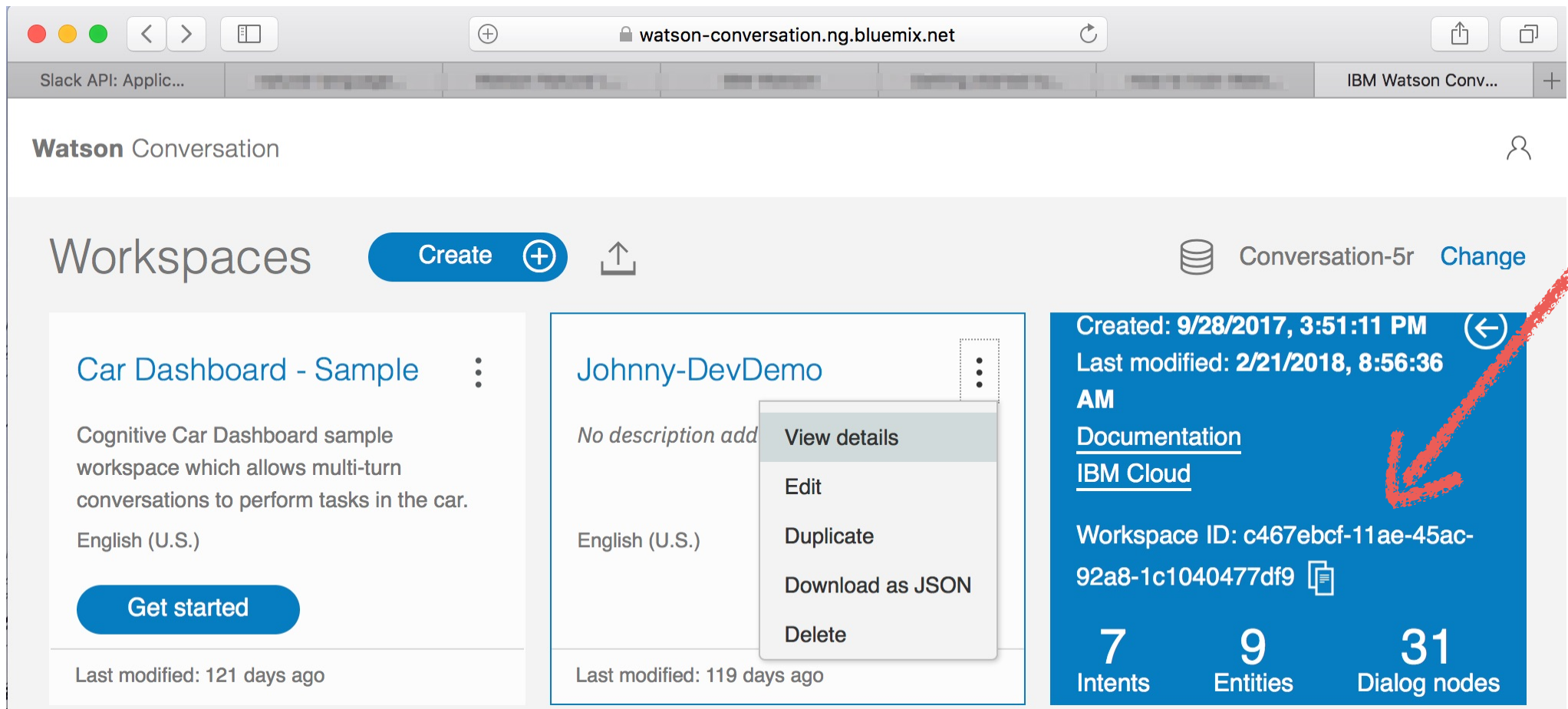
UNDERSTANDING

23



WATSON CONVERSATION

24



The screenshot shows the IBM Watson Conversation console interface. At the top, the browser address bar displays `watson-conversation.ng.bluemix.net`. The main header includes the 'Watson Conversation' logo and a user profile icon. Below the header, the 'Workspaces' section is visible, featuring a 'Create' button and an upload icon. Three workspace cards are shown: 'Car Dashboard - Sample', 'Johnny-DevDemo', and 'Conversation-5r'. The 'Johnny-DevDemo' card is highlighted, and its context menu is open, displaying options: 'View details', 'Edit', 'Duplicate', 'Download as JSON', and 'Delete'. A red arrow points to the 'View details' option. The 'Conversation-5r' card shows detailed information: 'Created: 9/28/2017, 3:51:11 PM', 'Last modified: 2/21/2018, 8:56:36 AM', 'Documentation', 'IBM Cloud', 'Workspace ID: c467ebcf-11ae-45ac-92a8-1c1040477df9', and statistics: 7 Intents, 9 Entities, and 31 Dialog nodes.

Watson Conversation

Workspaces [Create](#)

Car Dashboard - Sample

Cognitive Car Dashboard sample workspace which allows multi-turn conversations to perform tasks in the car.

English (U.S.)

[Get started](#)

Last modified: 121 days ago

Johnny-DevDemo

No description added

English (U.S.)

Last modified: 119 days ago

Conversation-5r [Change](#)

Created: 9/28/2017, 3:51:11 PM

Last modified: 2/21/2018, 8:56:36 AM

[Documentation](#)

[IBM Cloud](#)

Workspace ID: c467ebcf-11ae-45ac-92a8-1c1040477df9

7 Intents 9 Entities 31 Dialog nodes

WORKSPACE

25

The screenshot shows the IBM Watson Conversation workspace interface. The browser address bar displays `watson-conversation.ng.bluemix.net`. The page title is "Watson Conversation / Car Dashboard - Sample / Build". The left sidebar contains navigation icons for Intents, Entities, and Dialog. The main content area is titled "Intents" and shows a list of 25 intents, sorted by Name (A-Z). The list includes:

Intent Name	Description	Count
#about_VA	any music recommendation?	32
#capabilities	tell me what things you can do	100+
#compound_questions	radio on and lights off	36
#decision_replies	No thank you	10

On the right, a "Try it out" chat preview shows a sample conversation:

Try it out Clear Manage Context 9

Hi. It looks like a nice drive today. What would you like me to do?

Enter something to test your bot
Use the up key for most recent

INTENT EXAMPLE

26

The screenshot displays the Watson Conversation console. The left sidebar contains navigation icons. The main panel is titled 'Watson Conversation / Johnny - DevVersion / Build' and has tabs for 'Intents', 'Entities', and 'Dialog'. The 'Intents' tab is active, showing a list of 6 intents. The intents are:

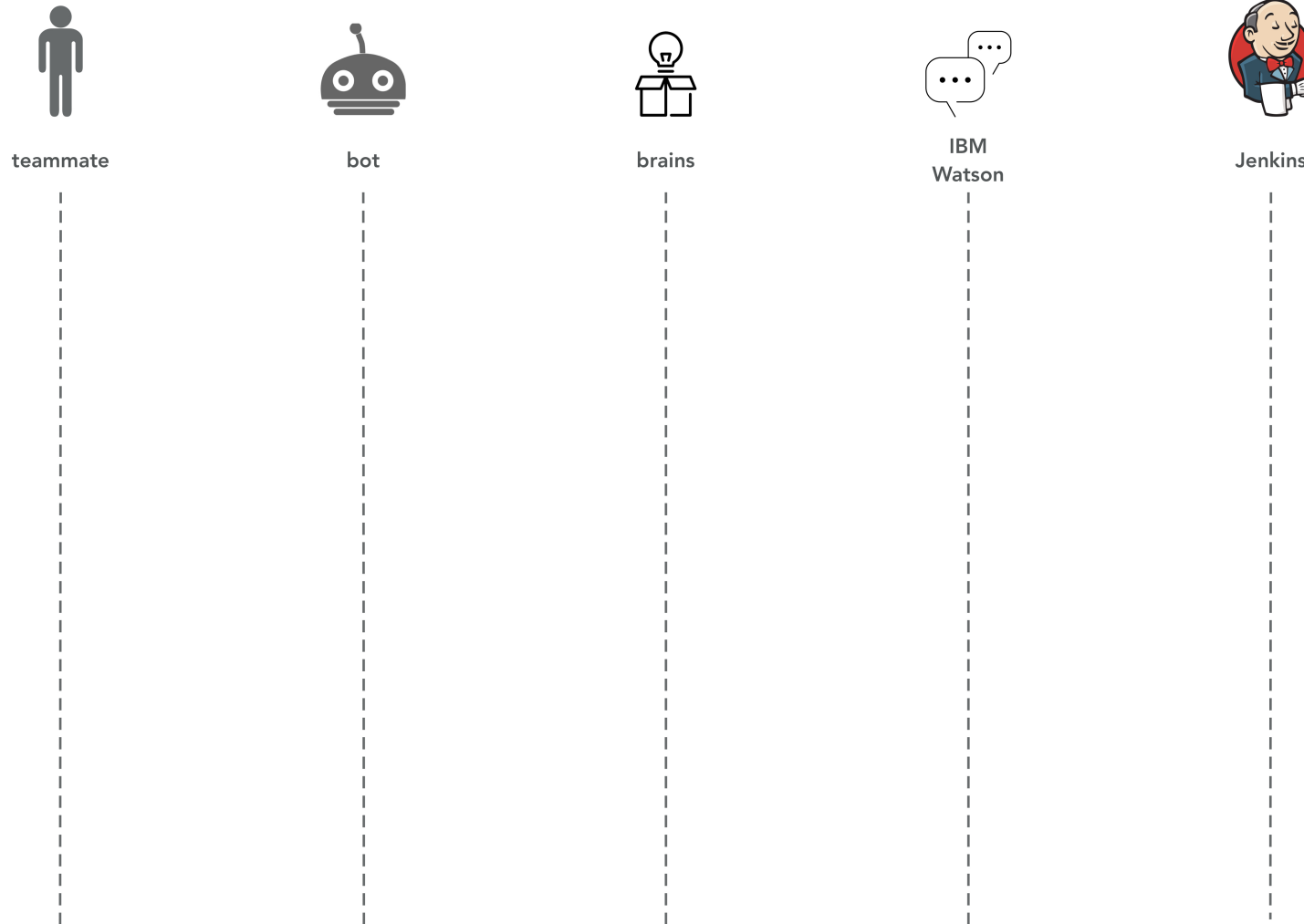
- #3_component_info** (3 examples): When did we last install the API in staging?
- #4_system_info** (3 examples): What was the traffic on the web server this week ?
- #1_tickets_ready_for_qa** (7 examples):
 - + Add a new user example...
 - ☐ Are there issues ready for testing?
 - ☐ What can we test ?
 - ☐ What jiras do we have ready for qa ?
 - ☐ Show me which tickets are ready for QA
 - ☐ List me the tickets ready for QAT
 - ☐ Tell me which jiras ready for testing ?
- #2_Prepare_RC_and_Deploy** (3 examples): Prepare an RC and deploy to stage
- #package** (7 examples):

On the right, a 'Try it out' chat window is open, showing a message: 'Hello. I am Johnny, how may I help you?' and a prompt to 'Enter something to test your bot'.

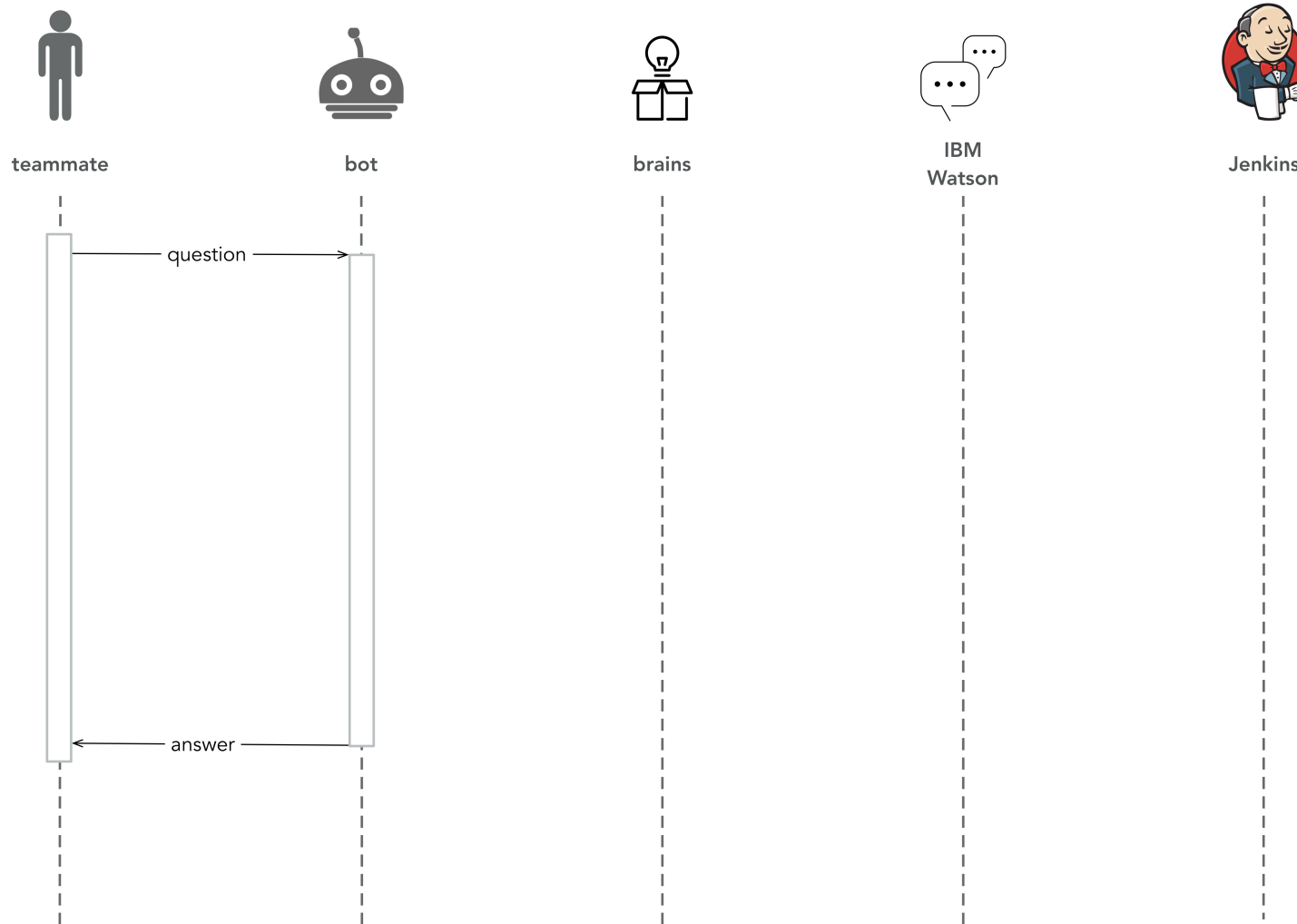


JOHNNY

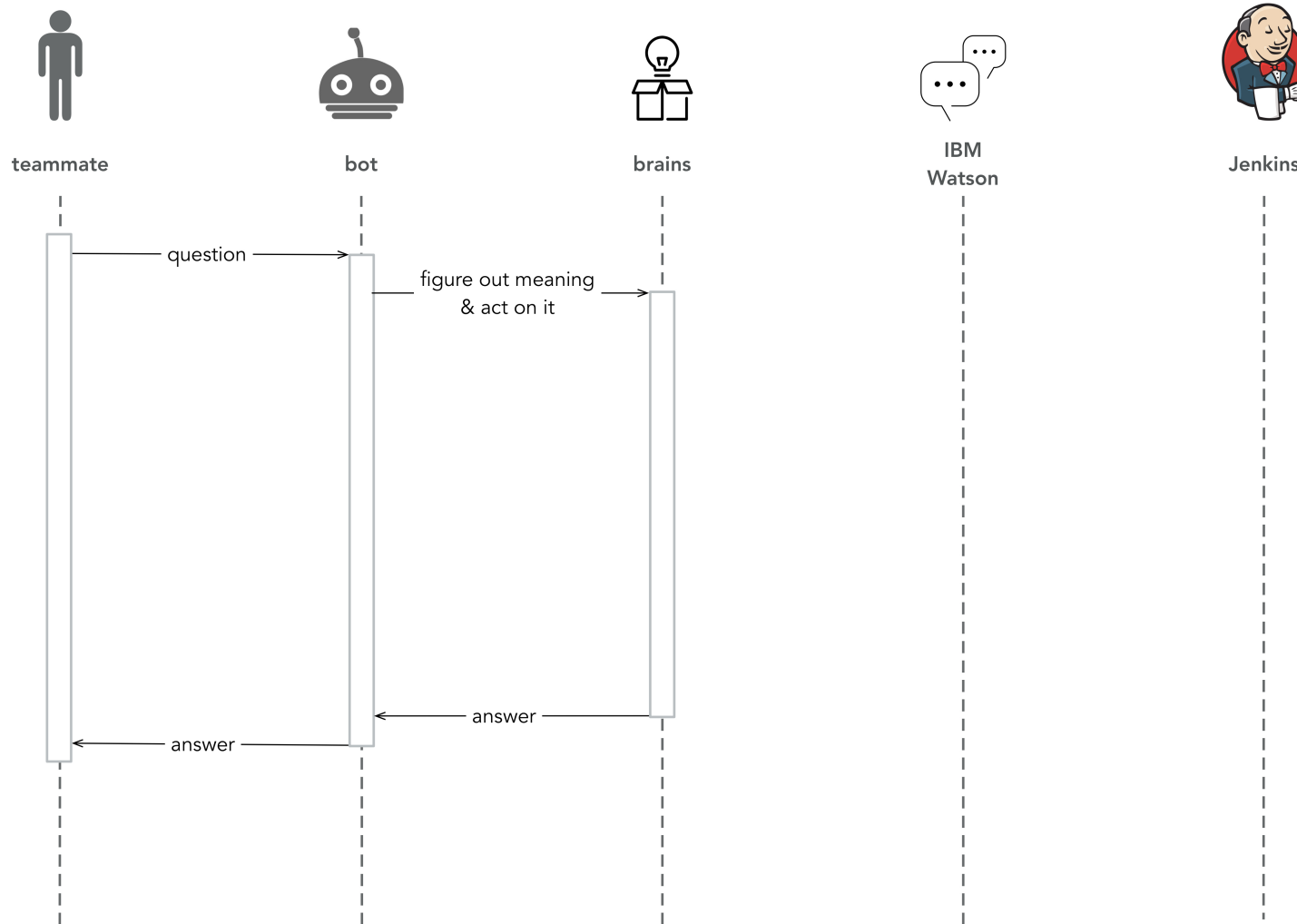
THE FLOW



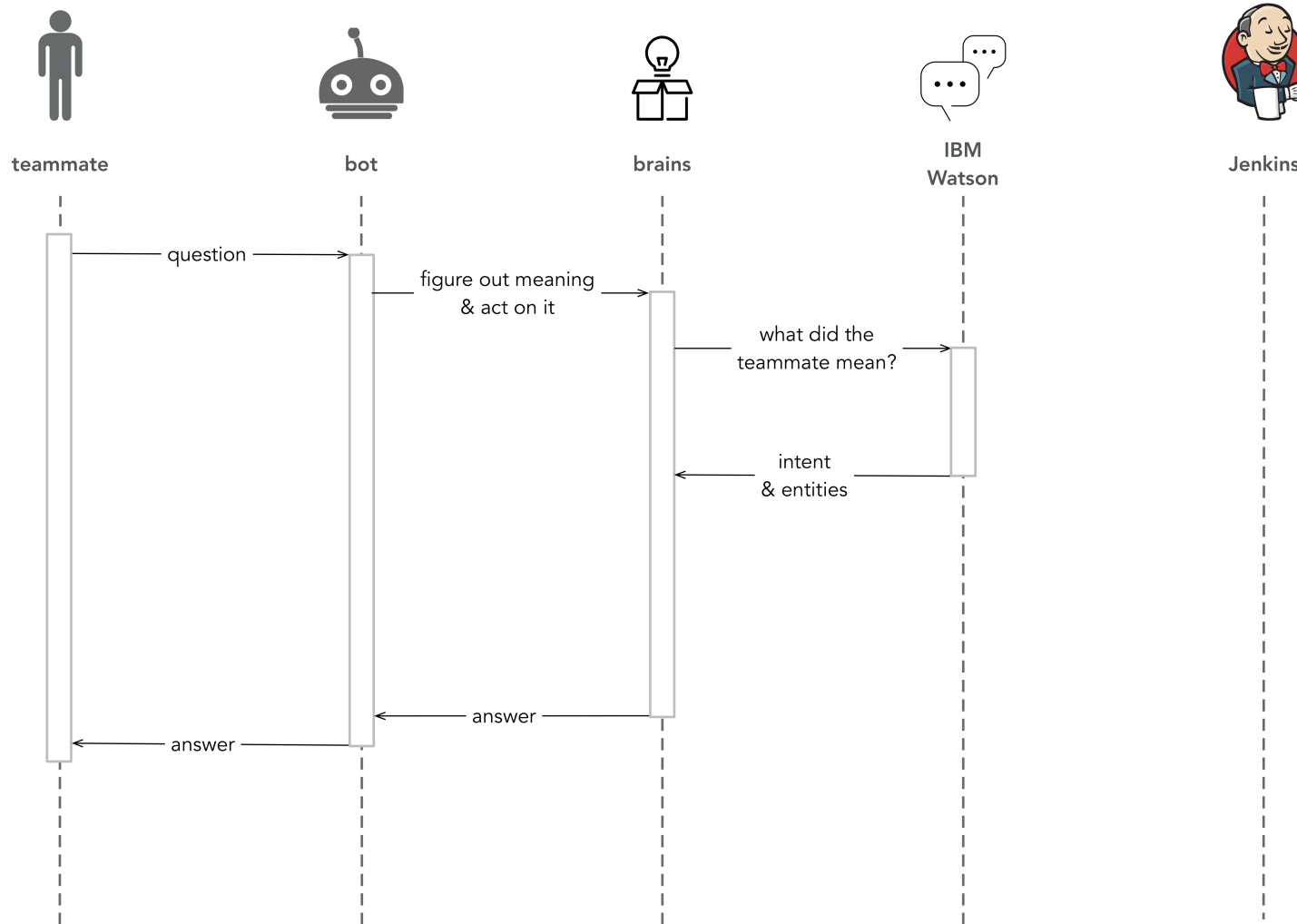
THE FLOW



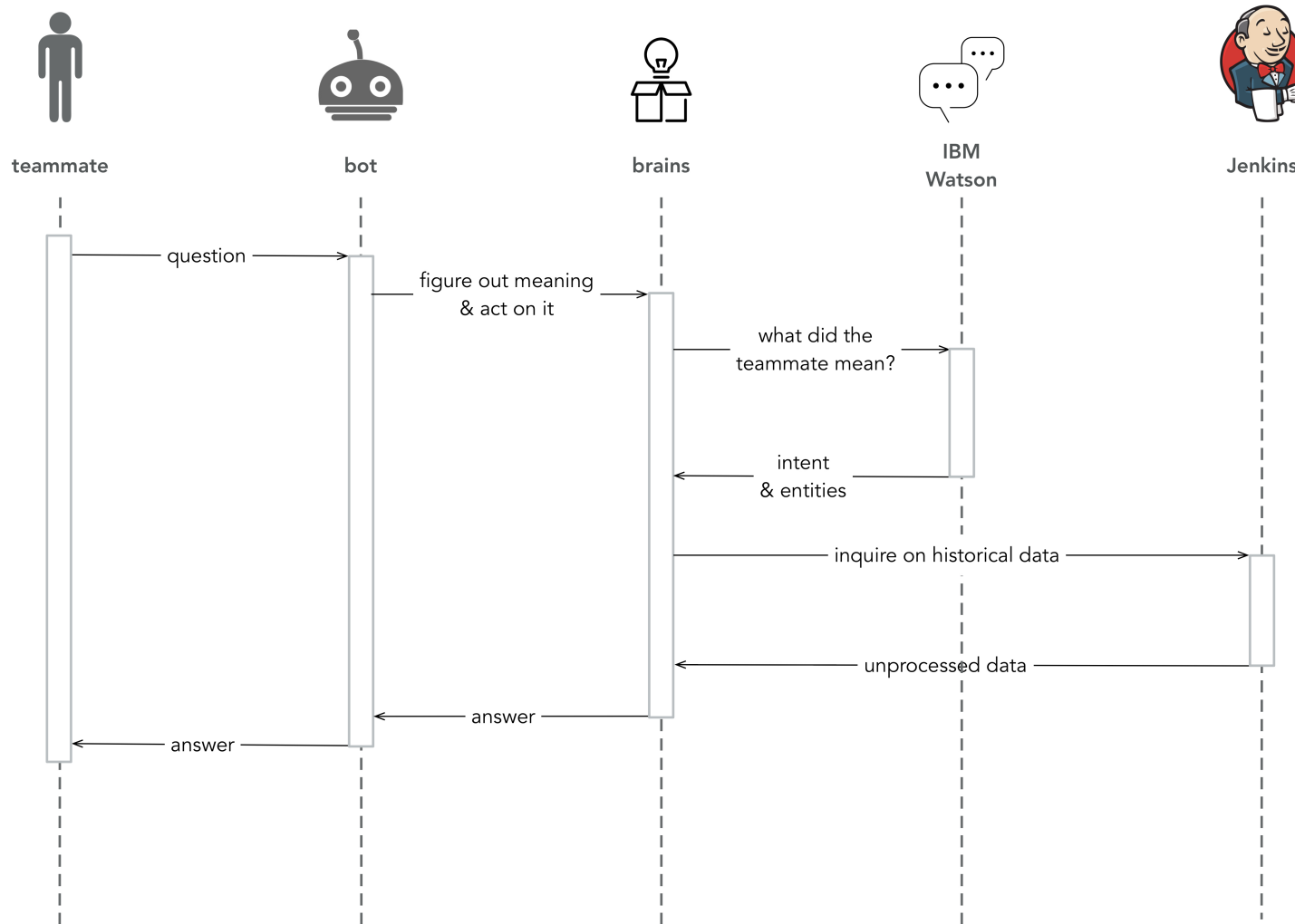
THE FLOW



THE FLOW

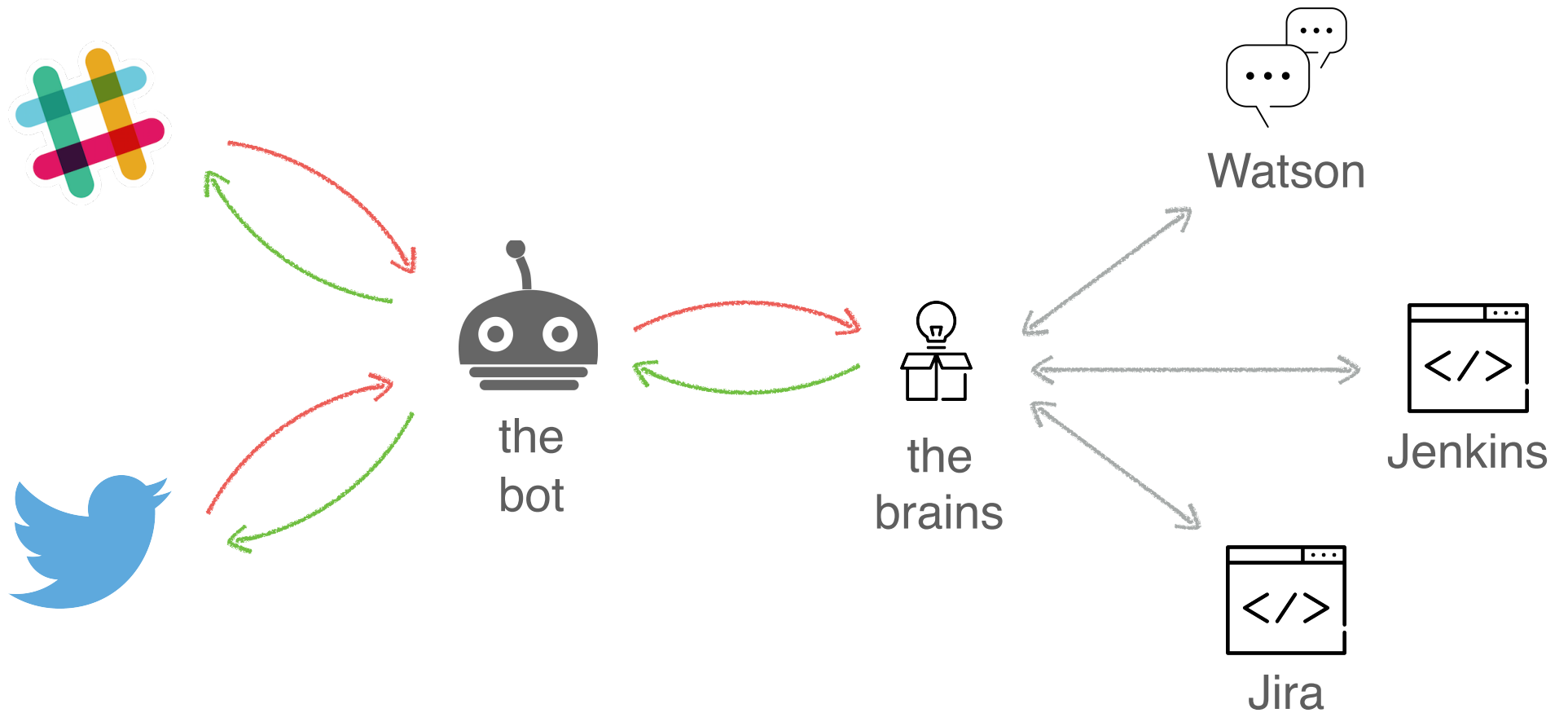


THE FLOW



OVERALL VIEW

33



SLACK

34

The screenshot shows the Slack API website with the 'Create a Slack App' dialog box open. The dialog has a title bar 'Create a Slack App' and a close button. Inside, there's a promotional banner for the next generation of apps. Below that, the 'App Name' field contains 'e.g. Super Service'. The 'Development Slack Workspace' dropdown menu is open, showing 'Development Slack Workspace' as the selected option, 'Tekkie Consulting' as another option, and a '+ Sign in to another workspace' button. At the bottom, there are 'Cancel' and 'Create App' buttons. A green box with the number '1' and an arrow points to the 'Development Slack Workspace' dropdown. Another green box with the number '2' and an arrow points to the 'Create App' button.

Slack API

Start here

- Building Slack apps
- Recent updates
- Best practices
- App blueprints

App features

- Internal integrations
- Incoming webhooks
- Slash commands
- Bot users
- Dialogs
- Shared Channels
- Enterprise Grid
- Legacy custom integrations
- Developer preview

Create a Slack App

Interested in the next generation of apps?
We're improving app development and distribution. Join the API Preview period for workspace tokens and the Permissions API.

App Name

e.g. Super Service

Don't worry; you'll be able to change this later.

Development Slack Workspace

Development Slack Workspace

Tekkie Consulting

+ Sign in to another workspace

Service.

Cancel Create App

[Documentation](#)[Tutorials](#)[Your Apps](#)

Johnny the Deployer ▼

Settings

[Basic Information](#)[Collaborators](#)[Install App](#)[Manage Distribution](#)

Features

[Incoming Webhooks](#)[Interactive Components](#)[Slash Commands](#)[OAuth & Permissions](#)[Event Subscriptions](#)[Bot Users](#)

Installed App Settings

OAuth Tokens for Your Team

These tokens were automatically generated when you installed the app to your team. You can use these to authenticate your app. [Learn more.](#)

OAuth Access Token

xoxp-5114068137-5114068149-260322045874-ea4bb1eef8ed8bc7cd3001ca72166 [Copy](#)

Bot User OAuth Access Token

xoxb-261206884230-ky2qub65LD8LIHJr1m3msBOv [Copy](#)

[Reinstall App](#)

1

2

 Johnny the Deployer ▼

Event Subscriptions

Settings

Basic Information
Collaborators
Install App
Manage Distribution

Features

Incoming Webhooks
Interactive Components
Slash Commands
OAuth & Permissions

Event Subscriptions

Enable Events

On ☒

Your app can subscribe to be notified of events in Slack (for example, when a user adds a reaction or creates a file) at a URL you choose. [Learn more.](#)

Request URL **Verified** ✓

`https://johnny-bot.ngrok.io/slack/johnny-hooked`

Change

We'll send HTTP POST requests to this URL when events occur. As soon as you enter a URL, we'll send a request with a `challenge` parameter, and your endpoint must respond with the challenge value. [Learn more.](#)

Subscribe to Bot Events

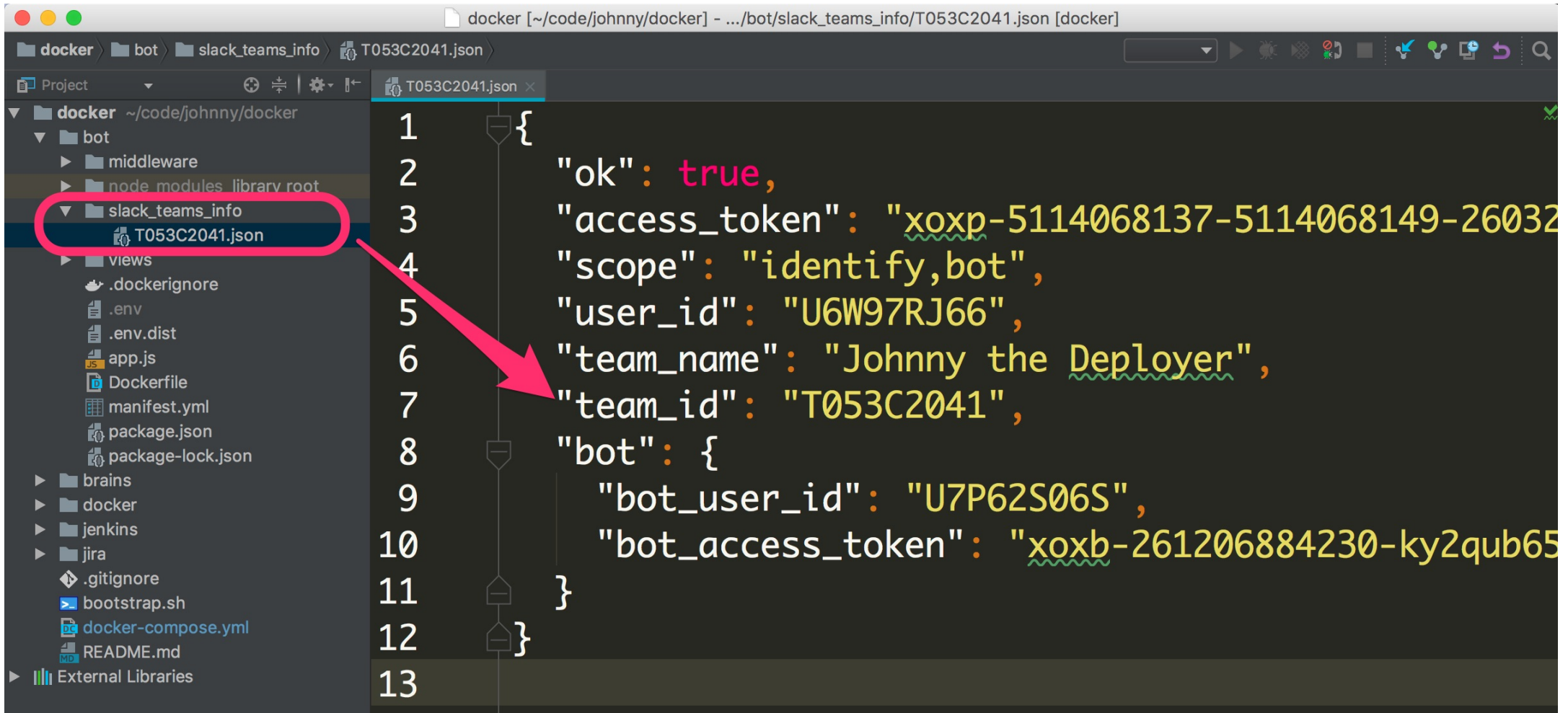
Bot users can subscribe to events related to the channels and conversations they're part of.

Event Name	Description	
message.channels	A message was posted to a channel	
message.groups	A message was posted to a private channel	
message.im	A message was posted in a direct message channel	
message.mpim	A message was posted in a multiparty direct message channel	

Add Bot User Event

BOTMASTER - SLACK

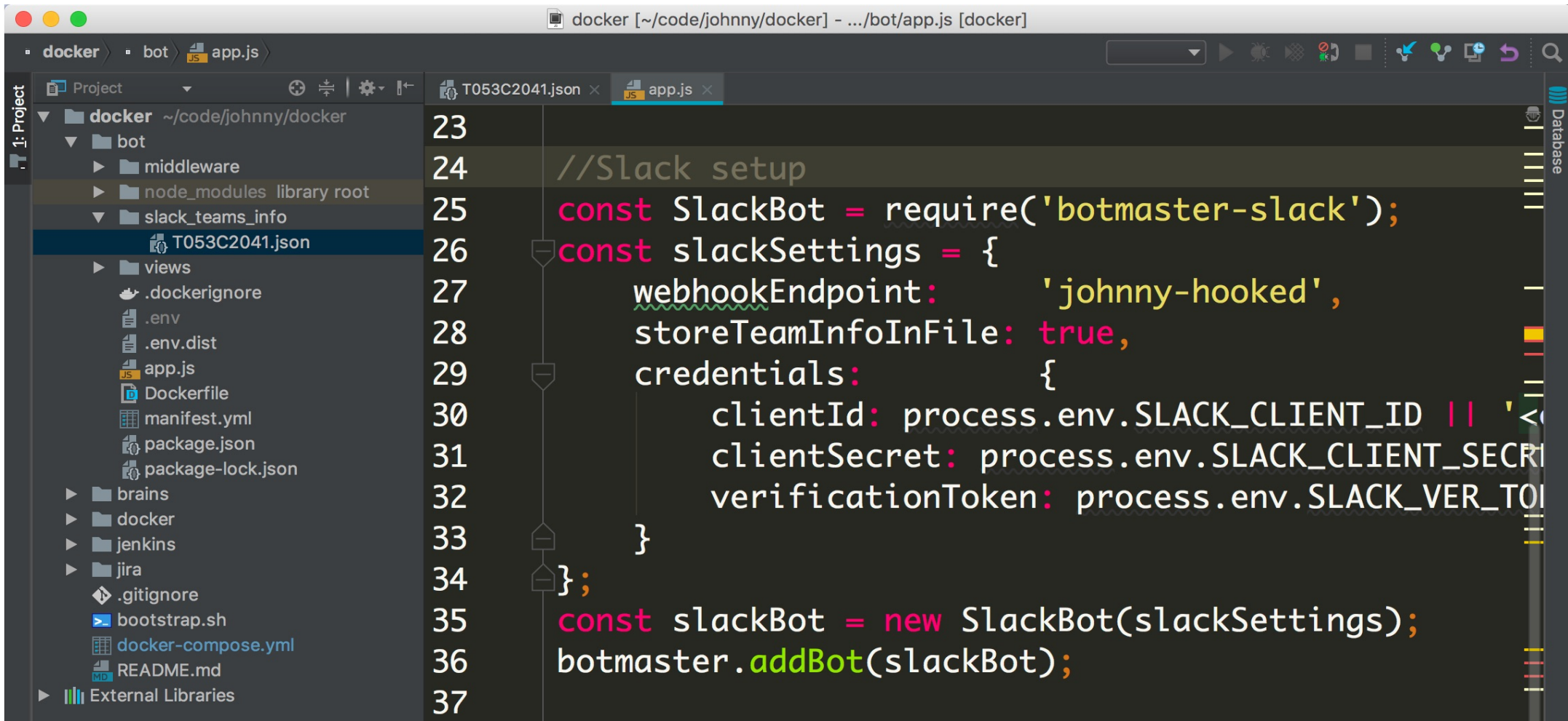
38



```
1  {
2    "ok": true,
3    "access_token": "xoxp-5114068137-5114068149-26032",
4    "scope": "identify,bot",
5    "user_id": "U6W97RJ66",
6    "team_name": "Johnny the Deployer",
7    "team_id": "T053C2041",
8    "bot": {
9      "bot_user_id": "U7P62S06S",
10     "bot_access_token": "xoxb-261206884230-ky2qub65",
11   }
12 }
13
```

BOTMASTER - SLACK

39



```
23
24 //Slack setup
25 const SlackBot = require('botmaster-slack');
26 const slackSettings = {
27   webhookEndpoint: 'johnny-hooked',
28   storeTeamInfoInFile: true,
29   credentials: {
30     clientId: process.env.SLACK_CLIENT_ID || '<
31     clientSecret: process.env.SLACK_CLIENT_SECRET,
32     verificationToken: process.env.SLACK_VERIFICATION_TOKEN
33   }
34 };
35 const slackBot = new SlackBot(slackSettings);
36 botmaster.addBot(slackBot);
37
```

BOTMASTER - SLACK

40

```
32 botmaster.use({
33   type: 'incoming',
34   name: 'slack-incoming-middleware',
35   controller: (bot, update) => {
36     if (typeof update.raw.event.bot_id !== 'undefined' || process.env.SLACK_BOT_USER_ID === update.raw.event.user) {
37       console.log('skipping ' + update.raw.event.text + ' with bot_id=' + update.raw.event.bot_id);
38       return;
39     }
40
41     var requestOptions = {
42       method: 'post',
43       url: process.env.BRAIN_ENDPOINT,
44       body: {
45         message: update.raw.event.text
46       },
47       json: true,
48       resolveWithFullResponse: true,
49       simple: false
50     };
51
52     requestLib(requestOptions)
53       .then(function(response) {
54         if (response.statusCode !== 200) {
55           return bot.reply(update, 'I encountered a technical issue that prevented me from understanding you.');
```

THE BRAINS

POST /

[Summary](#)[Headers](#)[Raw](#)[Binary](#)[Replay](#)

47 bytes application/json

```
{  
  "message": "when did we deploy the cms in qa?"  
}
```

200 OK

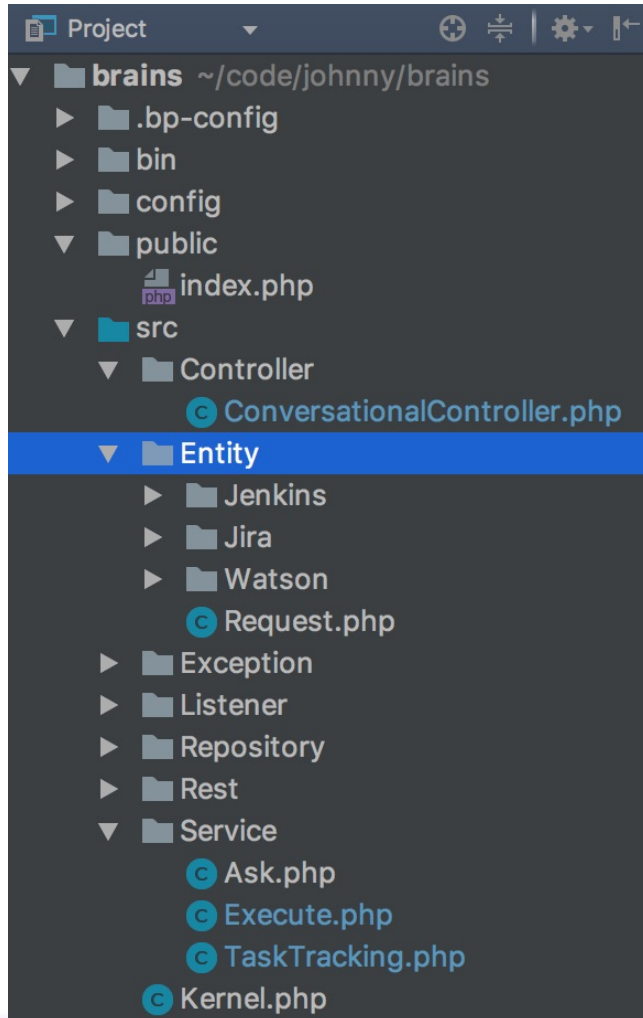
[Summary](#)[Headers](#)[Raw](#)[Binary](#)

191 bytes text/html; charset=UTF-8

```
Let me find out about: deploy of cms in qa. It was on 2  
017-10-26 12:31:27 and was marked as FAILURE. <https://  
devops.tekkie.ro/view/all/job/2\_Prepere\_RC\_and\_Deploy/2  
8/console|See the details>
```


THE BRAINS

42



```
11 class ConversationalController extends Controller
12 {
13
14     public function index(Request $request)
15     {
16         $responseText = '';
17
18         // decode what we got from JohnnyTheBot
19         $apiRequest = $this->get('jms_serializer')->deserialize(
20             $request->getContent(),
21             'App\Entity\Request',
22             'json'
23         );
24
25         if (is_null($apiRequest->message)) {
26             return new Response('');
27         }
28
29         /** @var \App\Service\Ask $ask */
30         $ask = $this->get('conversation_ask');
31         $apiResponse = $ask->ask($apiRequest->message);
32         $this->get('logger')->debug(sprintf('Watson says: %s', $this-
33
34         // append what Watson said to what we'll give back to botmaster
35         $responseText .= implode(' ', $apiResponse->getOutput()->text);
36
37         if (0 == count($apiResponse->getIntents())) {
38             $this->get('logger')->debug(sprintf(
39                 format: 'No intent for me this time. Watson said `%s`',
40                 $responseText
41             ));
42             return new Response($responseText);
43         }
44     }
45 }
```

THE BRAINS

```
45     foreach ($apiResponse->getIntents() as $intent) {
46         /** @var Intent $intent */
47         switch ($intent->getIntent()) {
48             case '1_tickets_ready_for_environment':
49                 $responseText .= $this->readyForEnvironment(
50                     $apiResponse->getEnvironment()
51                 );
52                 break;
53             case '2_Prepare_RC_and_Deploy':
54                 $this->buildAndDeploy(
55                     $apiResponse->getEnvironment(),
56                     $apiResponse->getComponent()
57                 );
58                 break;
59             case '3_component_info':
60                 $responseText .= $this->lastComponentActionOnEnvironment(
61                     $apiResponse->getComponent(),
62                     $apiResponse->getAction(),
63                     $apiResponse->getEnvironment()
64                 );
65                 break;
66             default:
67                 break;
68         }
69     }
70
71     return new Response($responseText);
72 }
```

THE BRAINS - ASK WATSON

44

```
44  /**
45   * Figure out Watson's intent from the provided question
46   *
47   * @param string $question
48   * @return \App\Entity\Watson\Response
49   */
50  public function ask(string $question): \App\Entity\Watson\Response
51  {
52      $request = new \App\Entity\Watson\Request($question);
53      $reqBody = $this->serializer->serialize($request, 'json');
54
55      $response = $this->watsonClient->request('post', '', ['body' => $reqBody]);
56
57      $this->logger->debug(sprintf('Watson said: `%s`', $response->getBody()));
58
59      /** @var \App\Entity\Watson\Response $deserialized */
60      $deserialized = $this->serializer->deserialize(
61          $response->getBody(),
62          'App\Entity\Watson\Response',
63          'json'
64      );
65
66      return $deserialized;
67  }
```


WATSON: COMPLEX INTENT

45

The screenshot displays the Watson Conversation management console for a project named 'Johnny - DevVersion'. The 'Intents' tab is active, showing a list of six intents. Each intent entry includes a chevron icon, a title, a description, and a count. To the right, a 'Try it out' dialog is open, showing a sample conversation with the bot.

Watson Conversation / Johnny - DevVersion / Build

Intents Entities Dialog

Create new (+) Import Export Delete 6 intents Sort by: Newest

Intent	Description	Count
#2_Prepare_RC_and_Deploy	I need you to deploy to integration a new RC	5
#3_component_info	When did we last install the API in staging?	3
#4_system_info	How did the traffic look for the last 36 hours?	3
#1_tickets_ready_for_qa	Show me which tickets are ready for QA	6
#package	make a build	7
#deploy	update integration	5

Try it out Clear Manage Context (1)

Hello. I am Johnny, how may I help you?

Enter something to test your bot
Use the up key for most recent

WATSON: CONTEXT VARIABLES

46

The screenshot displays the Watson Conversation 'Build' interface for a project named 'Johnny - DevVersion'. The 'Dialog' tab is active, showing a flow for '2. build & deploy'. The configuration includes an intent '#2_Prepare_RC_and_Deploy' with a response that sets the '@ENVIRONMENT' context variable to 'integration' and triggers the 'pack and deploy' action. A JSON snippet shows the context structure:

```
1 {
2   "context": {
3     "env": "@ENVIRONMENT",
4     "action": "@info_type"
5   },
6   "output": {
7     "text": {
8       "values": [
9         "pack and deploy"
10      ],
11      "selection_policy": "sequential"
12    }
13  }
14 }
```

On the right, the 'Try it out' panel shows a simulated conversation. The user input is 'pack and deploy', which triggers the 'pack and deploy' action and sets the '@ENVIRONMENT' context variable to 'integration'. The bot's response is 'I need you to deploy to integration a new R C', followed by the 'pack and deploy' action and the 'pack and deploy' response.

QUESTIONS?

Thank you goes to Georgiana Beju,
@gbtekkie



muricar@gmail.com