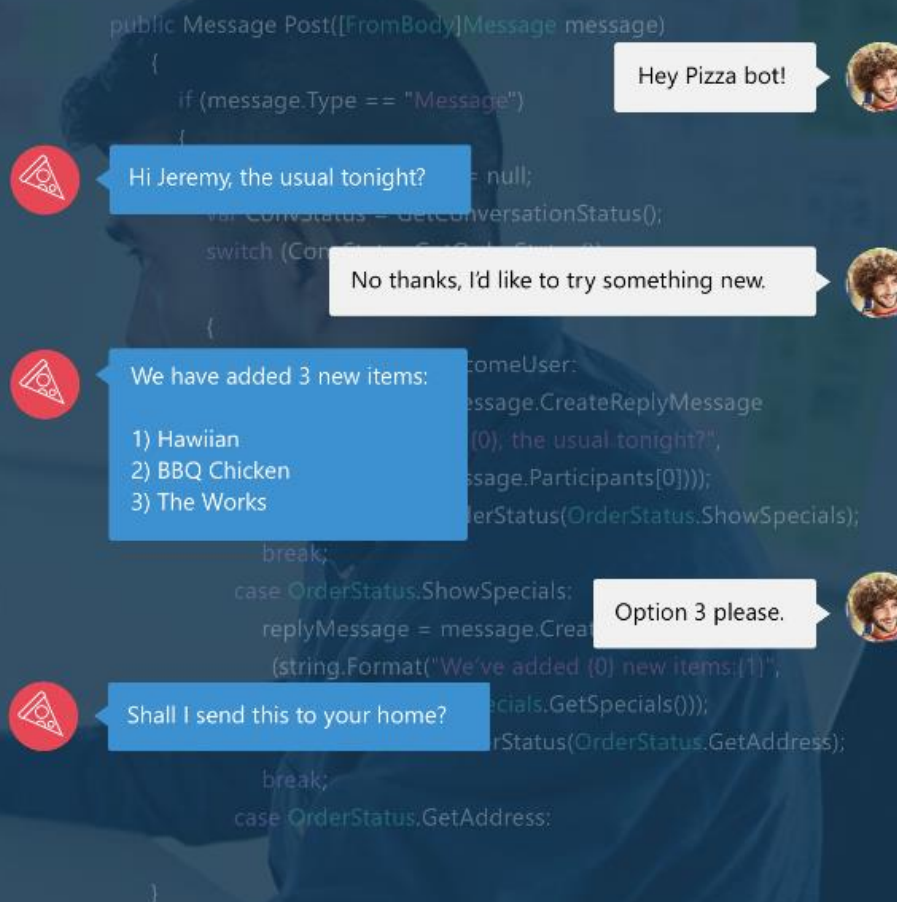


Microsoft Bot Framework

Your bots — wherever your users are talking.

Build and connect intelligent bots to interact with your users naturally wherever they are, from text/sms to Skype, Slack, Office 365 mail and other popular services.

Get started



```
public Message Post([FromBody]Message message)
{
    if (message.Type == "Message")
    {
        var ConversationStatus = null;
        var ConversationStatus = GetConversationStatus();
        switch (ConversationStatus)
        {
            case OrderStatus.ShowSpecials:
                var User = GetUser(message.Participants[0]);
                var ReplyMessage = message.CreateReplyMessage(
                    User, "We've added {0} new items: {1}",
                    message.Participants[0]));
                ConversationStatus(OrderStatus.ShowSpecials);
                break;
            case OrderStatus.ShowSpecials:
                replyMessage = message.CreateReplyMessage(
                    User, string.Format("We've added {0} new items: {1}",
                    message.Participants[0]));
                ConversationStatus(OrderStatus.GetAddress);
                break;
            case OrderStatus.GetAddress:
                // ...
            }
        }
    }
}
```

Hey Pizza bot!

Hi Jeremy, the usual tonight?

No thanks, I'd like to try something new.

We have added 3 new items:

- 1) Hawaiian
- 2) BBQ Chicken
- 3) The Works

Option 3 please.

Shall I send this to your home?

Bot Framework Components

Your Bot Framework Bot

Bot Connector

Connect your bot(s) to text/sms, Office 365 mail, Skype, Slack, Twitter, and other services.

- Register, connect, publish and manage your bot through the bot dashboard
- Message routing
- Automatic translation to 30+ languages
- User and state management
- Embeddable web chat control
- Debugging tools

Bot Builder SDKs

Build great dialogs within your Node.js- or C#-based bot

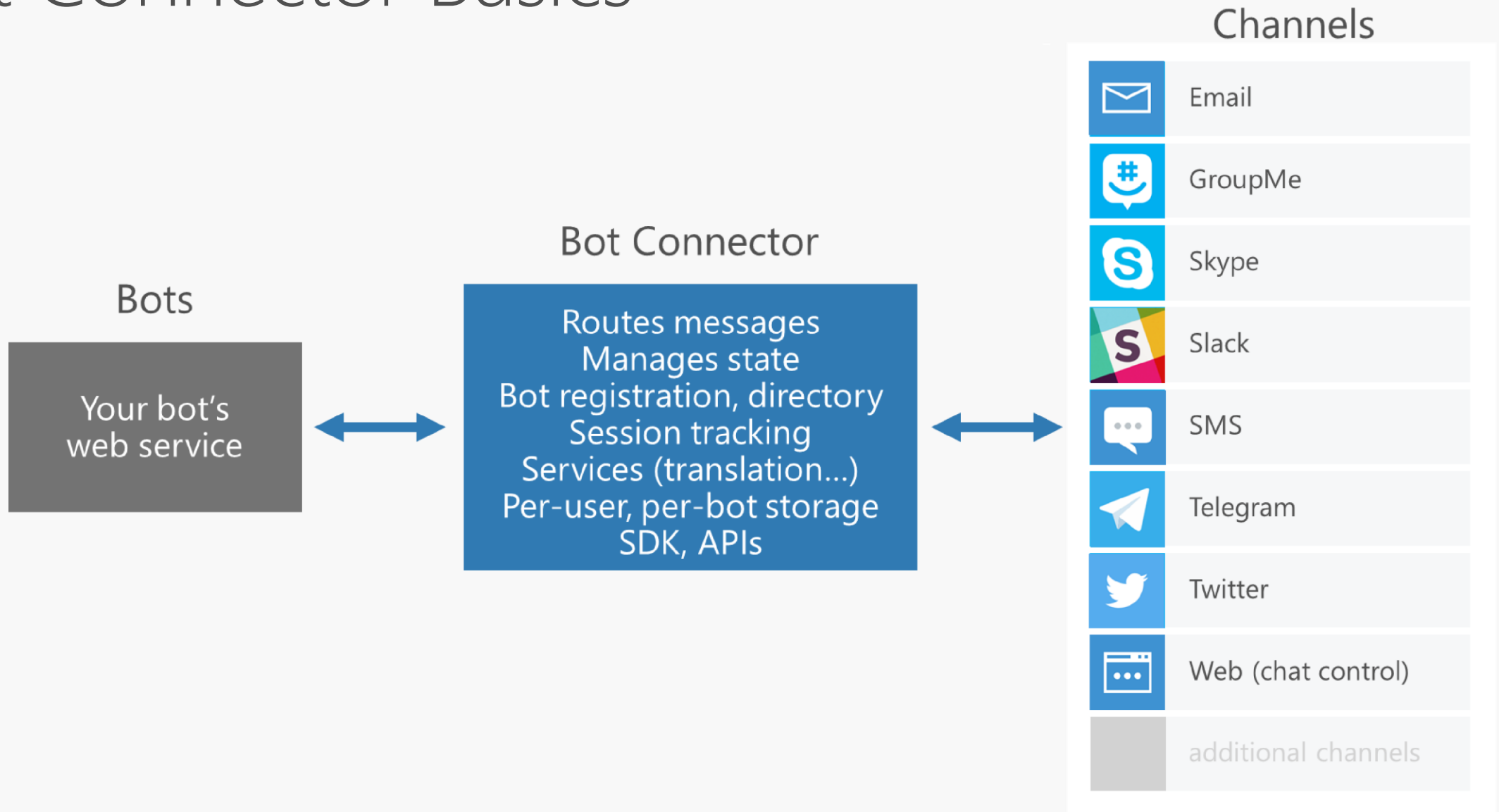
- Open source SDK on Github
- From simple built-in prompts and command dialogs to simple to use yet sophisticated 'FormFlow' dialogs
- Libraries, samples and tools to make a great conversationalist
- Chat emulator
- Leverage related services available in Cognitive Services

Bot Directory

COMING SOON - Try, use, and add published bots to the world's top conversation experiences

- Public directory of bots registered and approved with Bot Framework
- Users can try your bot from the directory via the web chat control
- Users can discover and add your bot to the channels on which it is configured

Bot Connector Basics



Developer Portal

- Register
- Configure Channels
- Test
- Manage

The screenshot shows the Microsoft Bot Framework Developer Portal interface for a bot named "Liberty Sandwich Bot". The top navigation bar includes links for "Bot framework", "My bots", "Register a bot", "Documentation", and "Bot directory", along with a search bar. The main content area is divided into three sections:

- Details:** Contains fields for App ID, Endpoint, Primary app secret, Secondary app secret, Listen to all messages (OFF), Translate channel messages (ON), Disable logging (OFF), and Publish in Bot directory (ON). Each field has a "Show" or "Edit" link.
- Channels:** A table showing the status of various channels. The "Web Chat" channel is currently "On", while "Email" is "Off". There are "Test link", "Status", and "Published" columns, and an "Edit" link for each channel.
- Add another channel:** A list of available channels including Direct Line, GroupMe, Skype, Slack, SMS, Telegram, and Twitter, each with an "Add" button.

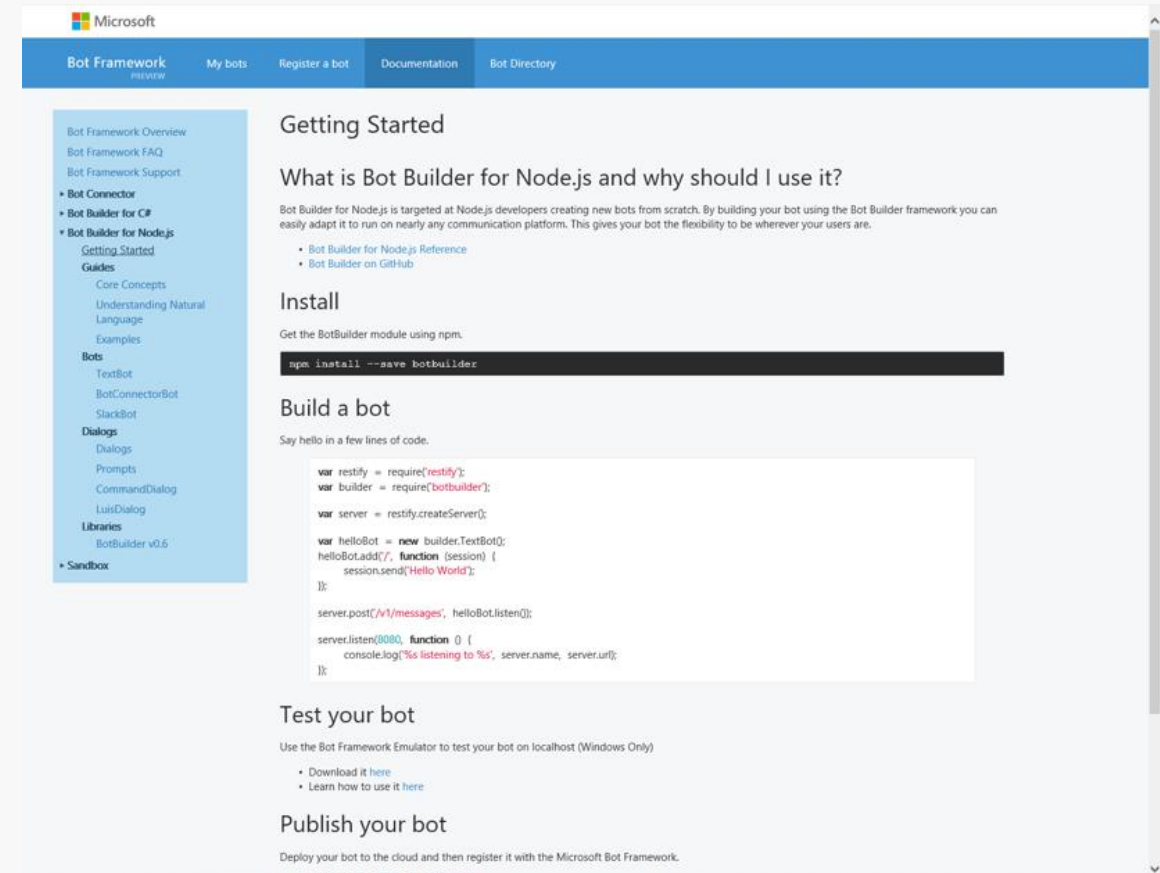
At the bottom, there is a "Test connection to Bot framework" section with a text input field and a "Send" button. A modal window titled "Configure Skype" is overlaid on the right side, showing instructions on how to connect the bot to Skype and a button to "Go to next step".

<https://dev.botframework.com/>

Bot Builder SDKs

C# and Node.js

- Dialogs to model a conversation
 - Dialogs are reusable
 - Conversations are scalable to multiple machines
 - Dialog state is persisted in Bot Connector (per-user, per-session, per- user+session)
- Types of Dialogs:
 - Built-in prompts
 - Yes/No, String, Number, Choices
 - Natural language understanding (LUIS)
 - Form flow

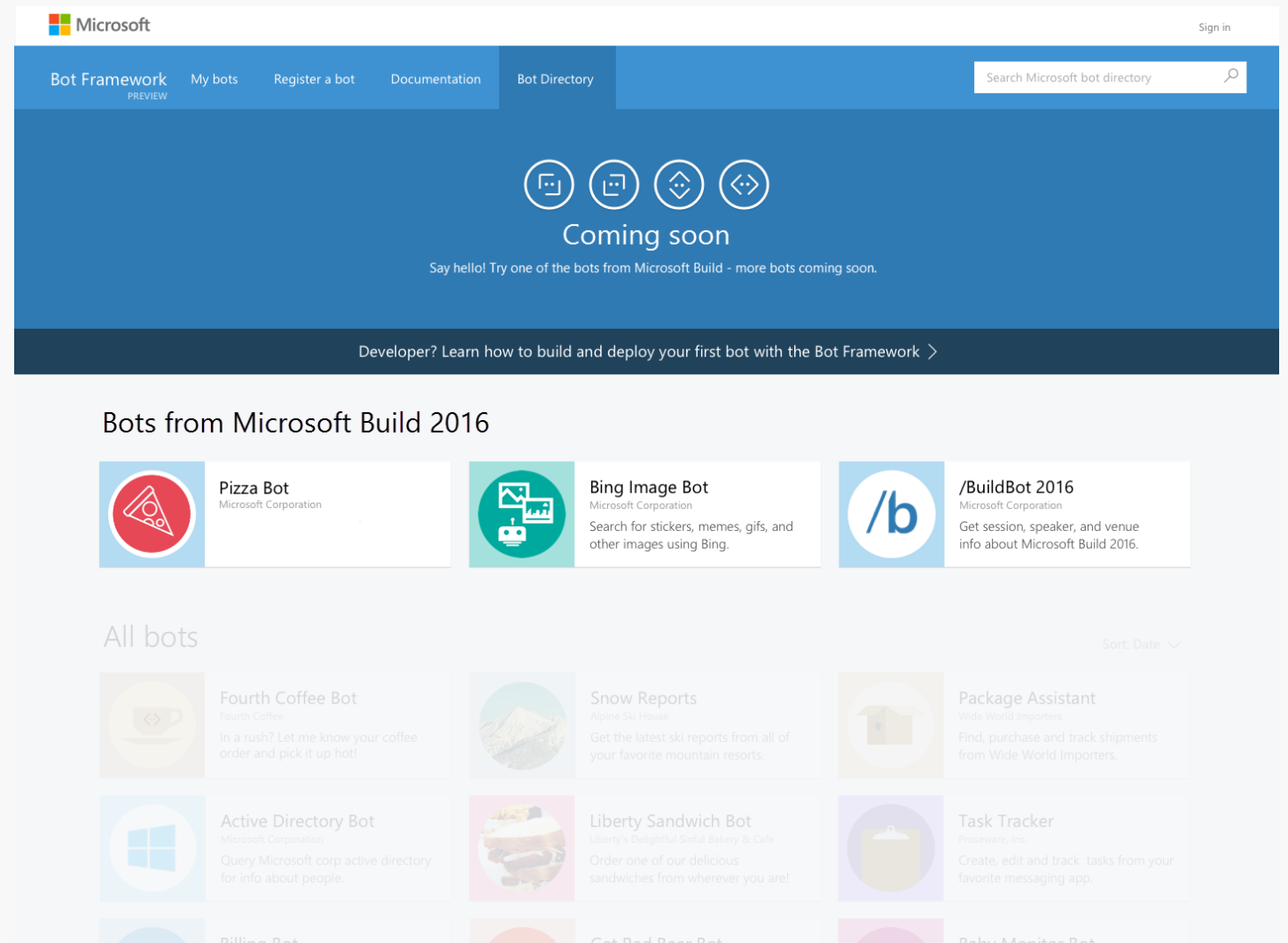


```
case OrderStatus.ShowSpecials:
    replyMessage = message.CreateReplyMessage
    (string.Format("We've added {0} new items:{1}",
    Specials.Count(), Specials.GetSpecials()));
    ConvStatus.SetOrderStatus(OrderStatus.GetAddress);
    break;
case OrderStatus.GetAddress:
}
```

Bot Directory (coming soon)

Public Directory of Bot Framework Bots

- Users can discover, try, and add bots to the conversation experiences on which the bot is configured (no app required)
- Bots are public at developer discretion; bots must be submitted for review and approved in order to appear in the directory
- Searchable



Benefits

- For developers
 - Easiest way to reach the broadest set of users where they already are conversing
 - Bots are more capable because of supporting services (translation, profile, history, memory, etc.)
 - Bring your own bot or build your own bot with the Bot Builder SDKs
- For end users
 - Users can choose from a variety of conversation channels
 - Users have trust and control of their data
- For businesses
 - Broad access to their customers, new experiences
 - Reduced cost of development



//re.build/

Demo code at:

<https://github.com/Danielius1012/ExampleBot>

Where to go from here?

```
"Recommended": [  
  {  
    "Title": "Building a Conversational Bot: From 0 to 60",  
    "Content": " https://channel9.msdn.com/Events/Build/2016/B821 "  
  },  
  {  
    "Title": "Here's How You Can Build Your Own Smart Bot",  
    "Content": " https://channel9.msdn.com/Events/Build/2016/T679-R1 "  
  },  
  {  
    "Title": "Bot Framework Documentation",  
    "Content": " http://docs.botframework.com/ "  
  }  
]
```

```
{  
  "About me": {  
    "Name": "Daniel Heinze",  
    "Mail": "t-dahein@microsoft.com",  
    "Description": "Software Developer since 2006",  
    "Twitter": " https://twitter.com/starlord\_daniel ",  
    "Facebook": " https://www.facebook.com/danielsdevblog/ ",  
    "Website": " http://devblog.azurewebsites.net/ "  
  }  
}
```



Thank you!