MVA Microsoft Virtual Academy

Lighting Up Real-time Web Communications with SignalR

Introduction

Microsoft

Jon Galloway | Technical Evangelist
Brady Gaster | Program Manager, Azure SDK & Tools

Meet Jon Galloway | @jongalloway

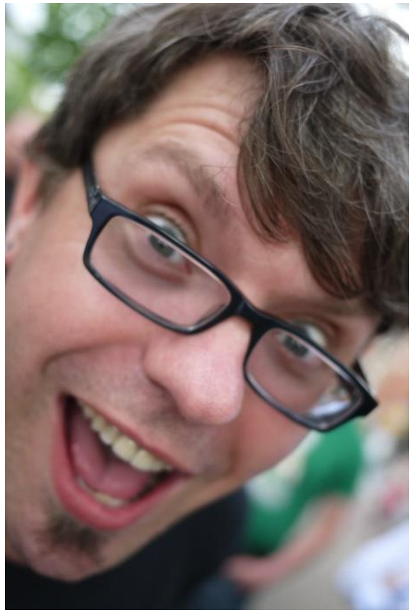
Azure Technical Evangelist

- Focused on ASP.NET MVC
- http://weblogs.asp.net/jongalloway
- Web development on Microsoft platform since late '90s
- Ex-submariner; Showcase Showdown winner “Price is Right”

Popular Author and Conference Speaker

- Wrox Professional MVC 5; MVC Music Store tutorial
- Virtual ASP.NET MVC Conference (mvcConf)
- World wide Web Camps speaker
- Herding Code podcast (http://herdingcode.com)

Meet Brady Gaster | @bradygaster



Program Manager, Azure SDK & Web Tools

- Focused on ASP.NET, Azure, and Visual Studio Web Tools
- http://www.bradygaster.com
- Web development on Microsoft platform since late '90s
- Musician, Father of 2 boys, NETMF tinkerer, ex-teacher

Speaker, Host, Blogger, Author

- Co-host of Channel 9’s Web Camps TV show
- Spoken at TechEd, //build/, and worldwide Web Camps events
- Azure Blog author http://blog.azure.com
- Organized AzureConf 2012 & 2013

Course Topics

Lighting Up Real-time Communications with SignalR

01 | Introduction to SignalR

02 | SignalR on the Web

03 | SignalR on the Client

04 | Scaling out SignalR

05 | Q&A and Advanced Demos with the SignalR Team

Setting Expectations

- Target Audience
  - HTML/.NET/JavaScript Developer
  - Basic knowledge of HTTP
  - Looking to fill knowledge gaps
- Suggested Prerequisites/Supporting Material
  - Visual Studio 2013 Express for Web

Join the MVA Community!

- Microsoft Virtual Academy
  - Free online learning tailored for IT Pros and Developers
  - Over 2.3M registered users
  - Up-to-date, relevant training on variety of Microsoft products
- “Earn while you learn!”
  - Get 50 MVA Points for this event!
  - Visit http://aka.ms/MVA-Voucher
  - Enter this code: 2WayCommSignalR (expires 10-24-2014)

MVA Microsoft Virtual Academy

01 | Introduction to SignalR

Microsoft

Jon Galloway | Technical Evangelist
Brady Gaster | Program Manager, Azure SDK & Tools

Any sufficiently advanced technology is indistinguishable from magic – Arthur C. Clarke

What is SignalR?

SignalR is a series of abstractions around various methods of providing persistent HTTP connections distributed as open-source code

[Can I get that in English?]

SignalR makes real-time HTTP so easy it seems like magic

SignalR Core Concepts

Connection

- Represents a simple endpoint for sending single-recipient, grouped, or broadcast messages

Hub

- A more high-level pipeline built upon the Connection API that allows your client and server to call methods on each other directly

Backplane

- A backplane allows you to scale your application to multiple servers. With a backplane enabled, each application instance sends messages to the backplane, and the backplane forwards them to the other application instances.

MVA Microsoft Virtual Academy

Installing SignalR and Getting Started

Microsoft

(we promise to minimize the use of chat demos today!)

MVA Microsoft Virtual Academy

DEMO

Creating a Real-time Hit Counter

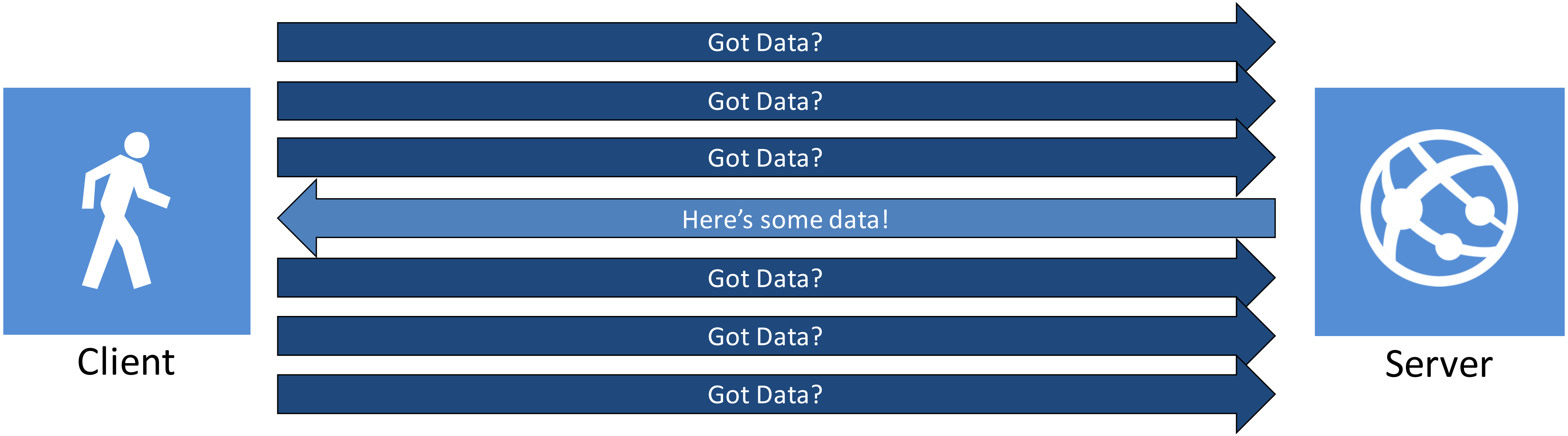
No applets or controls required!

MVA Microsoft Virtual Academy

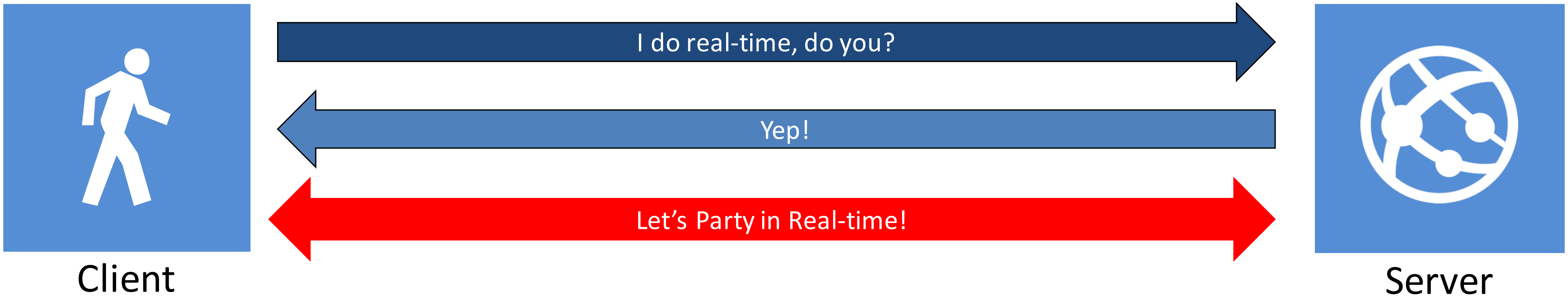
SignalR – How it Works

Microsoft

SignalR on Older Servers or Clients



SignalR on Modern Servers or Clients



I do real-time, do you?

Yep!

Let’s Party in Real-time!

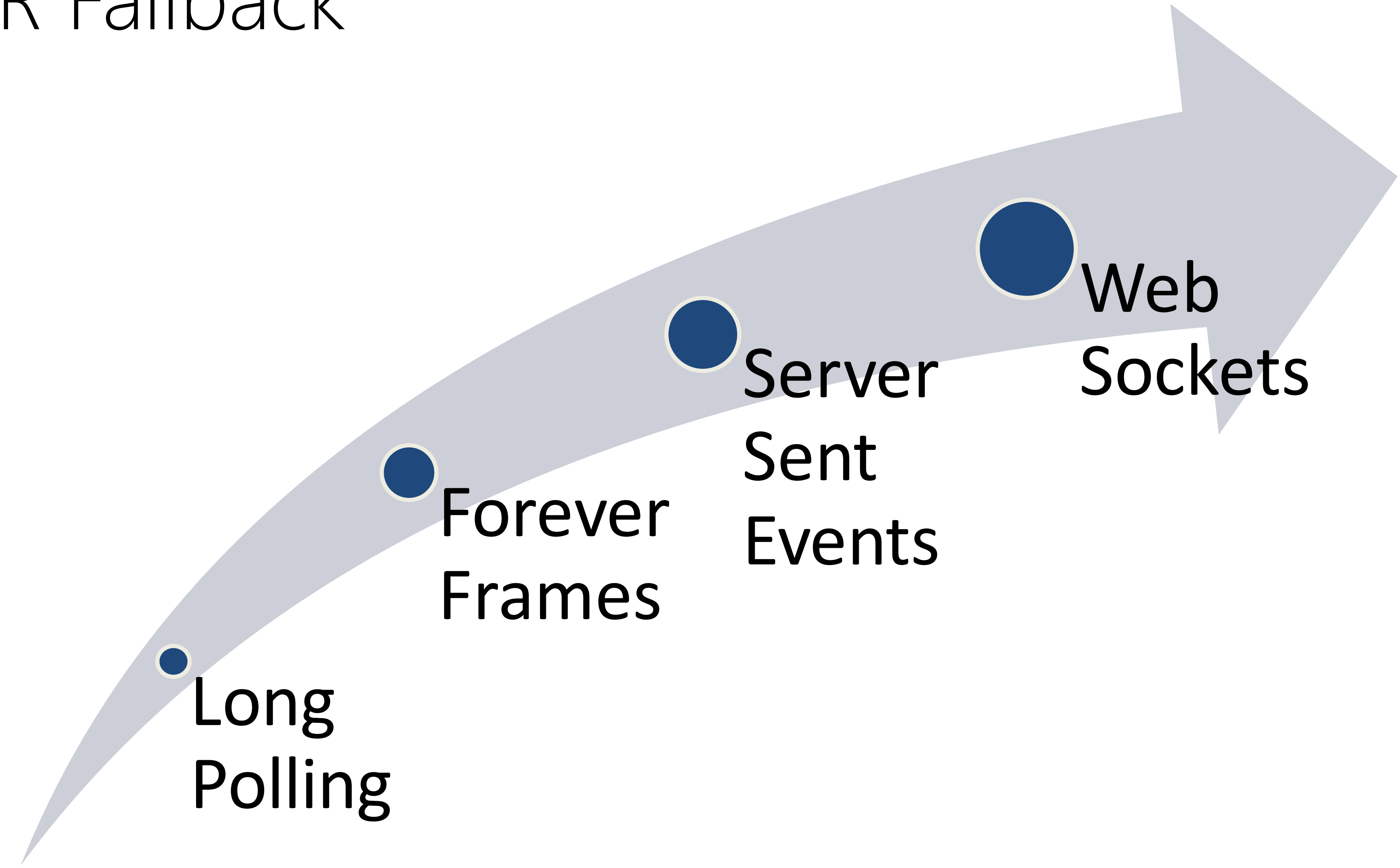
DEMO

How support for WebSockets changes network behavior

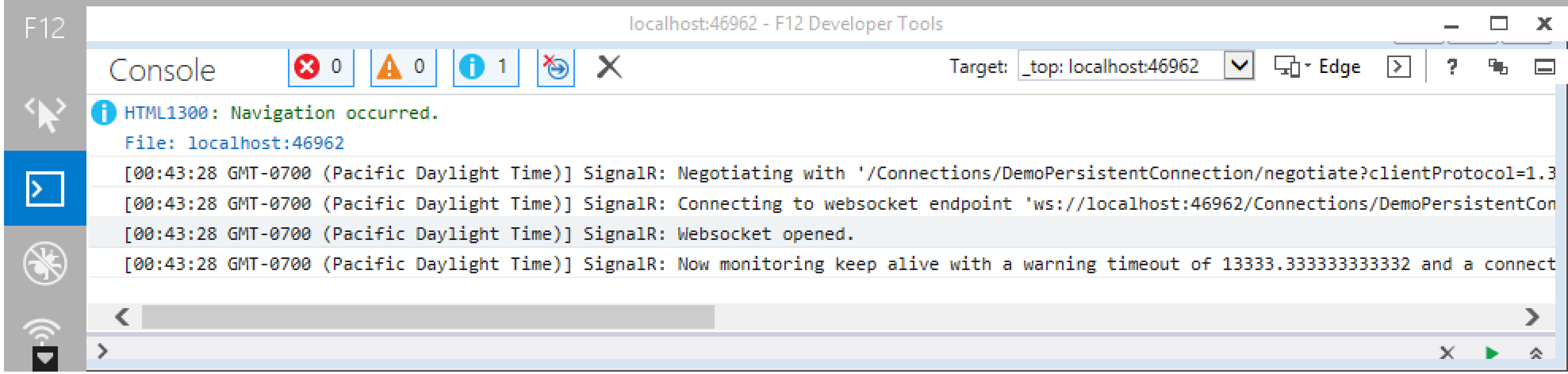
So how can SignalR know which methodology to use on both sides of the *persistent HTTP connection?

* ..and don’t think this drastic overhaul in your expectations of HTTP is abnormal. Everyone freaks out the first time they see it.

SignalR Fallback

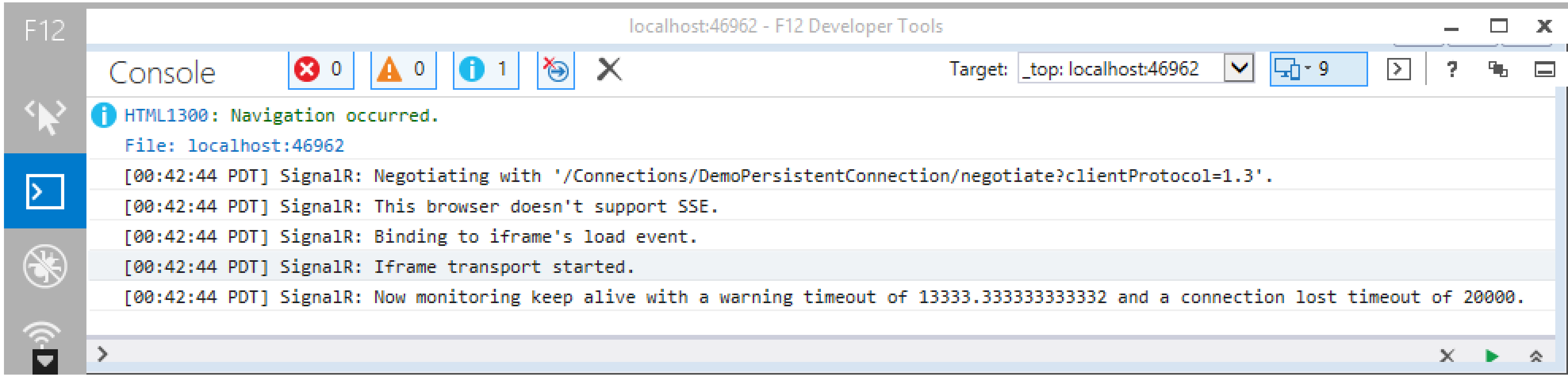


Let’s try
Websockets!



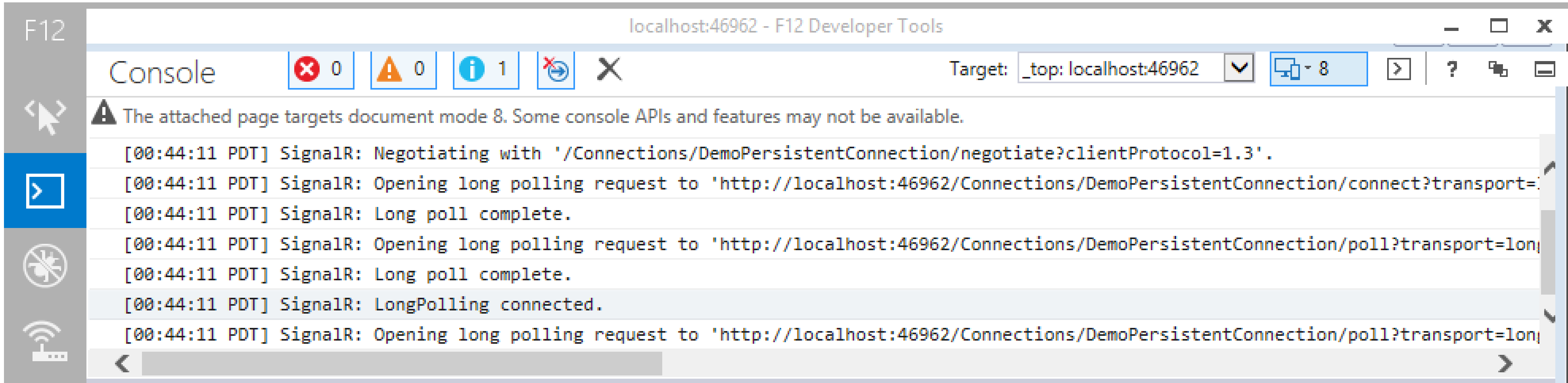
```
HTML1300: Navigation occurred.
File: localhost:46962
[00:43:28 GMT-0700 (Pacific Daylight Time)] SignalR: Negotiating with '/Connections/DemoPersistentConnection/negotiate?clientProtocol=1.3
[00:43:28 GMT-0700 (Pacific Daylight Time)] SignalR: Connecting to websocket endpoint 'ws://localhost:46962/Connections/DemoPersistentCon
[00:43:28 GMT-0700 (Pacific Daylight Time)] SignalR: Websocket opened.
[00:43:28 GMT-0700 (Pacific Daylight Time)] SignalR: Now monitoring keep alive with a warning timeout of 13333.333333333332 and a connect
```

No?
How about SSE?
No?
Forever Frames?



```
HTML1300: Navigation occurred.
File: localhost:46962
[00:42:44 PDT] SignalR: Negotiating with '/Connections/DemoPersistentConnection/negotiate?clientProtocol=1.3'.
[00:42:44 PDT] SignalR: This browser doesn't support SSE.
[00:42:44 PDT] SignalR: Binding to iframe's load event.
[00:42:44 PDT] SignalR: Iframe transport started.
[00:42:44 PDT] SignalR: Now monitoring keep alive with a warning timeout of 13333.333333333332 and a connection lost timeout of 20000.
```

Okay, then.
Long Polling!



The attached page targets document mode 8. Some console APIs and features may not be available.

```
[00:44:11 PDT] SignalR: Negotiating with '/Connections/DemoPersistentConnection/negotiate?clientProtocol=1.3'.
[00:44:11 PDT] SignalR: Opening long polling request to 'http://localhost:46962/Connections/DemoPersistentConnection/connect?transport=
[00:44:11 PDT] SignalR: Long poll complete.
[00:44:11 PDT] SignalR: Opening long polling request to 'http://localhost:46962/Connections/DemoPersistentConnection/poll?transport=lon
[00:44:11 PDT] SignalR: Long poll complete.
[00:44:11 PDT] SignalR: LongPolling connected.
[00:44:11 PDT] SignalR: Opening long polling request to 'http://localhost:46962/Connections/DemoPersistentConnection/poll?transport=lon
```

Jon, add slide(s) here walking the customer through the way SignalR determines what the client (and if possible, the server) does to up/downgrade the connection based on client/server capabilities

See here: https://github.com/SignalR/SignalR/blob/master/src/Microsoft.AspNet.SignalR.Client.JS/jquery.signalR.core.js

MVA Microsoft Virtual Academy

Persistent SignalR Connections

Microsoft

Persistent SignalR Connections

- Low level
- Work on top of transport abstraction
- Pass untyped messages using Send() and Broadcast()
- Client must parse message and understand what to do

Choosing a communication model

Most applications should use the Hubs API. The Connections API could be used in the following circumstances:

- The format of the actual message sent needs to be specified.
- The developer prefers to work with a messaging and dispatching model rather than a remote invocation model.
- An existing application that uses a messaging model is being ported to use SignalR.

http://www.asp.net/signalr/overview/signalr-20/getting-started-with-signalr-20/introduction-to-signalr

DEMO

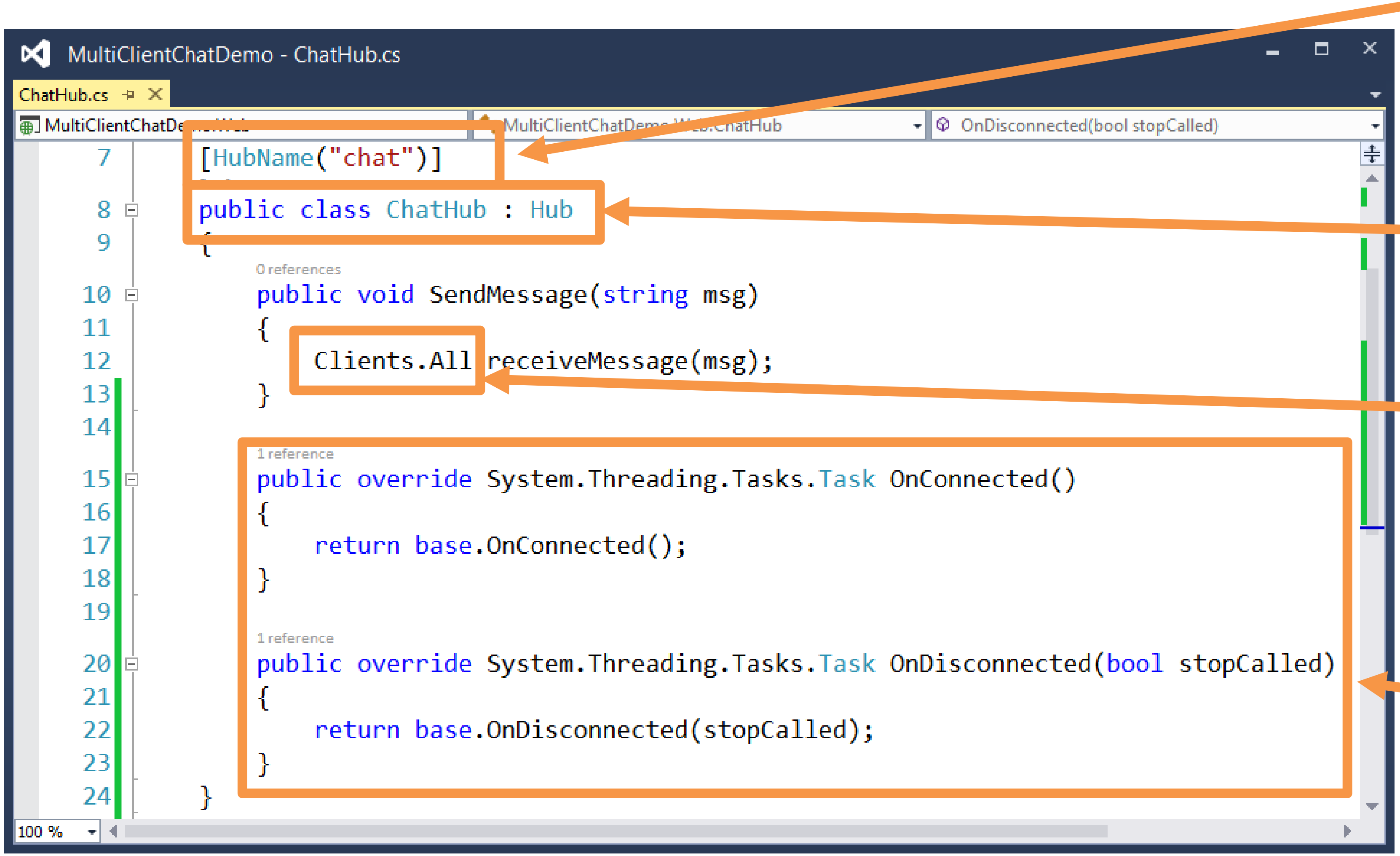
Persistent Connections

MVA Microsoft Virtual Academy

SignalR Hubs

Microsoft

SignalR Hubs



MultiClientChatDemo - ChatHub.cs
ChatHub.cs
MultiClientChatDe...
MultiClientChatDemo.Web.ChatHub
OnDisconnected(bool stopCalled)
100 %

```
[HubName("chat")]
public class ChatHub : Hub
{
    0 references
    public void SendMessage(string msg)
    {
        Clients.All.receiveMessage(msg);
    }

    1 reference
    public override System.Threading.Tasks.Task OnConnected()
    {
        return base.OnConnected();
    }

    1 reference
    public override System.Threading.Tasks.Task OnDisconnected(bool stopCalled)
    {
        return base.OnDisconnected(stopCalled);
    }
}
```

Since Hubs are called on the client by name, the name can be customized if needed

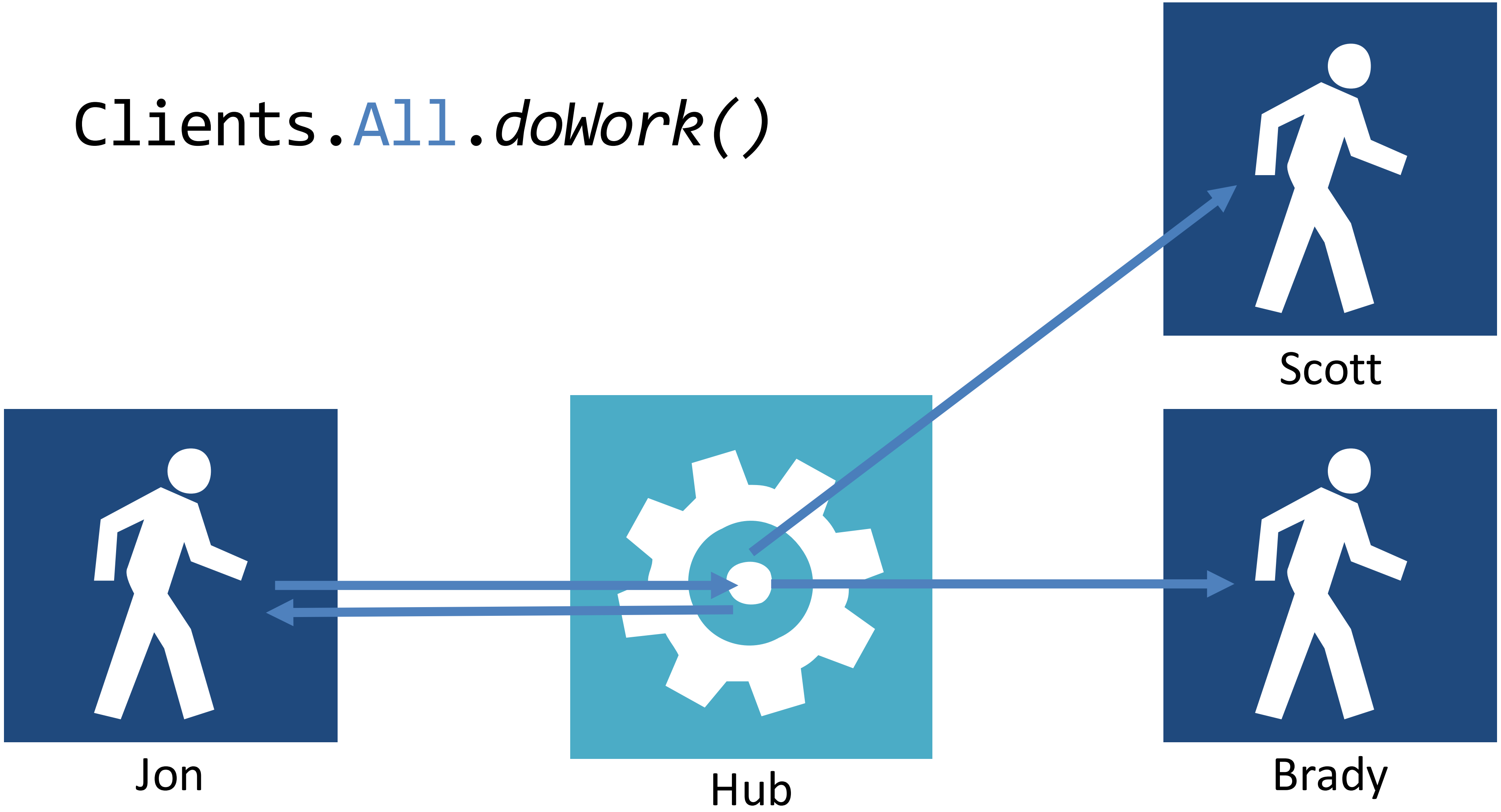
A hub is a .NET class that inherits from Microsoft.AspNet.SignalR.Hub

The Clients property of a Hub class exposes dynamic properties useful for targeting specific clients

Hub classes also have virtual methods useful for responding to connected/disconnected events

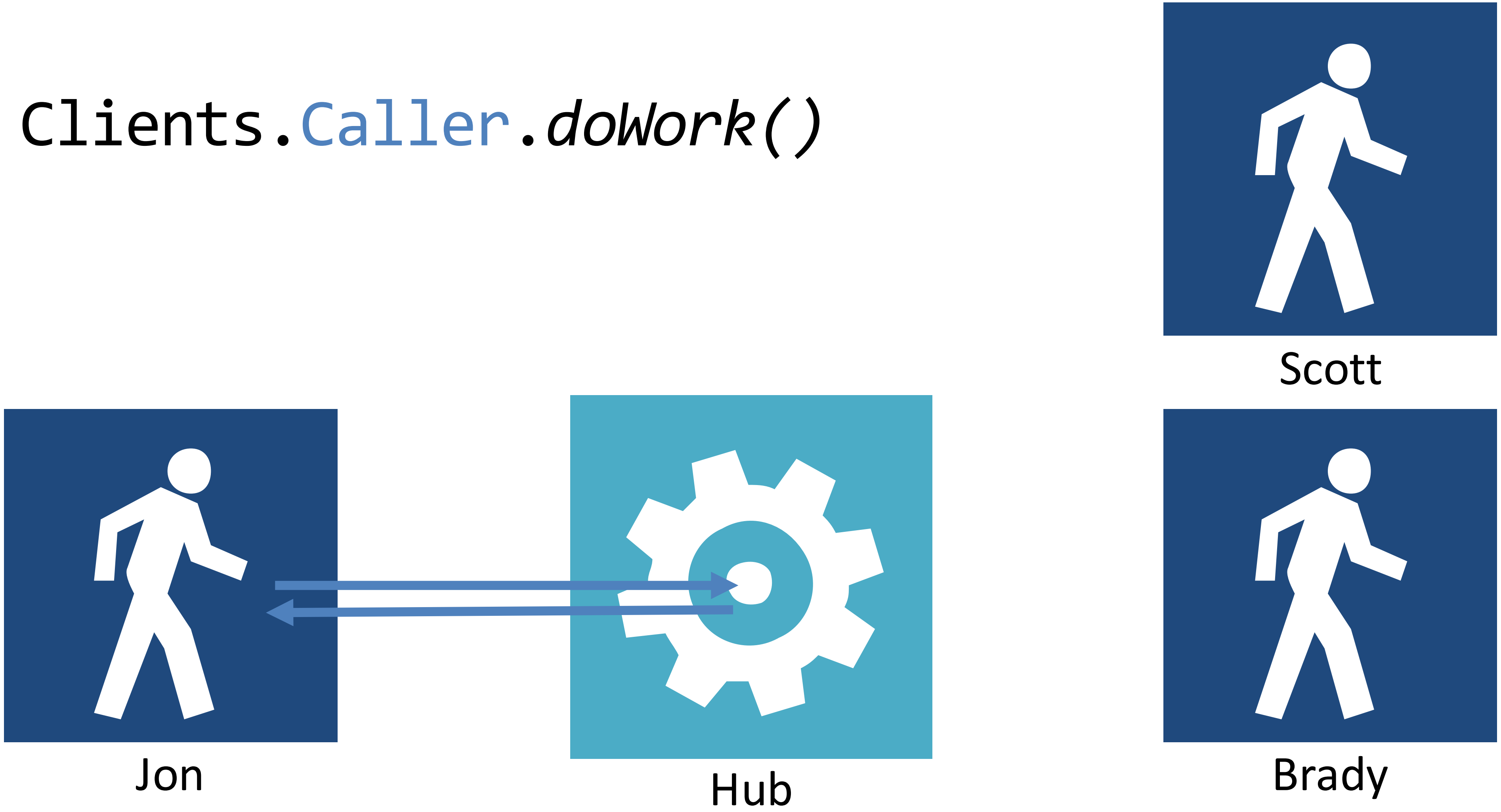
Sending messages to all connected clients

```
Clients.All.doWork()
```



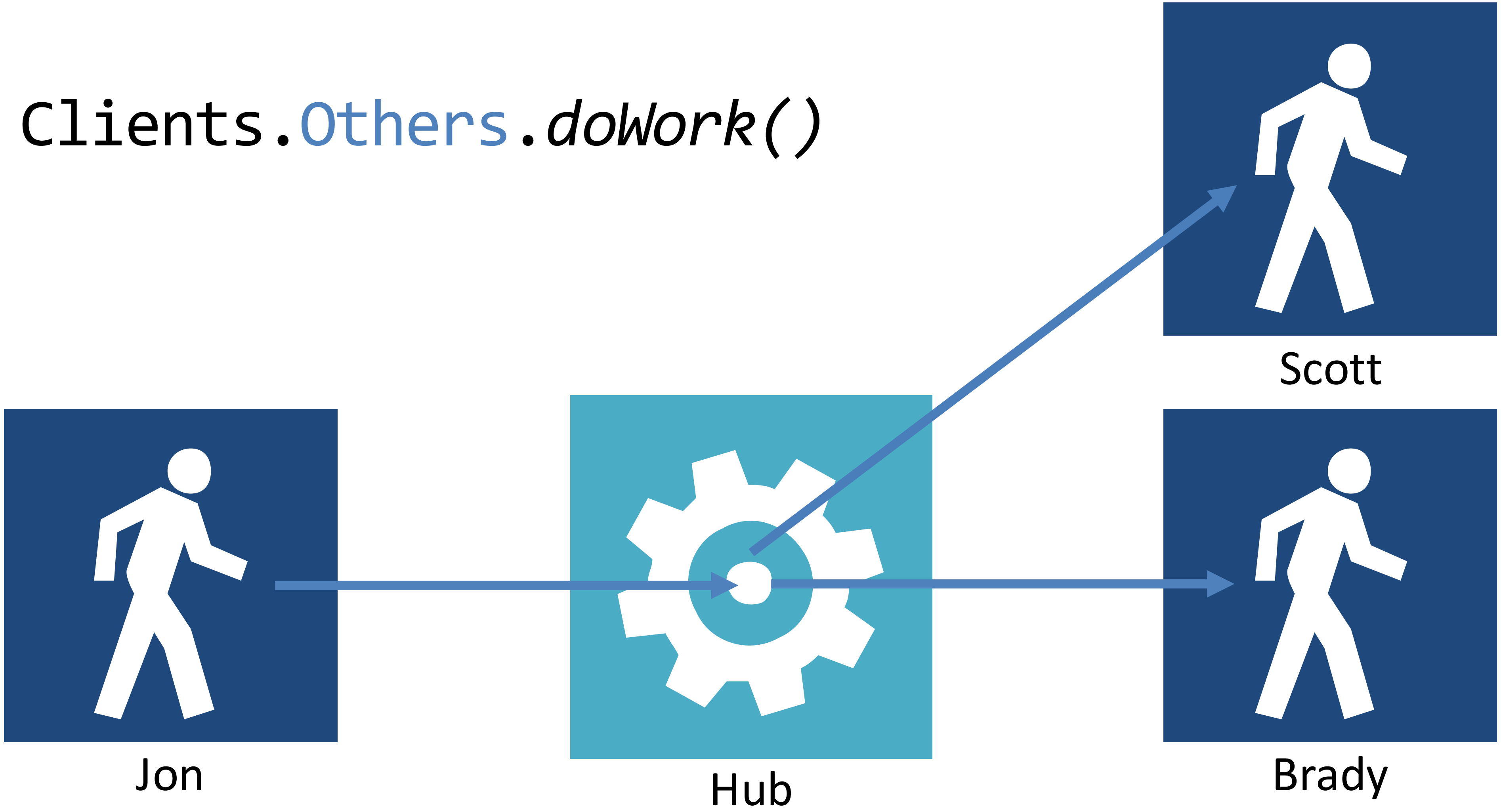
Sending messages to the calling client

```
Clients.Caller.doWork()
```



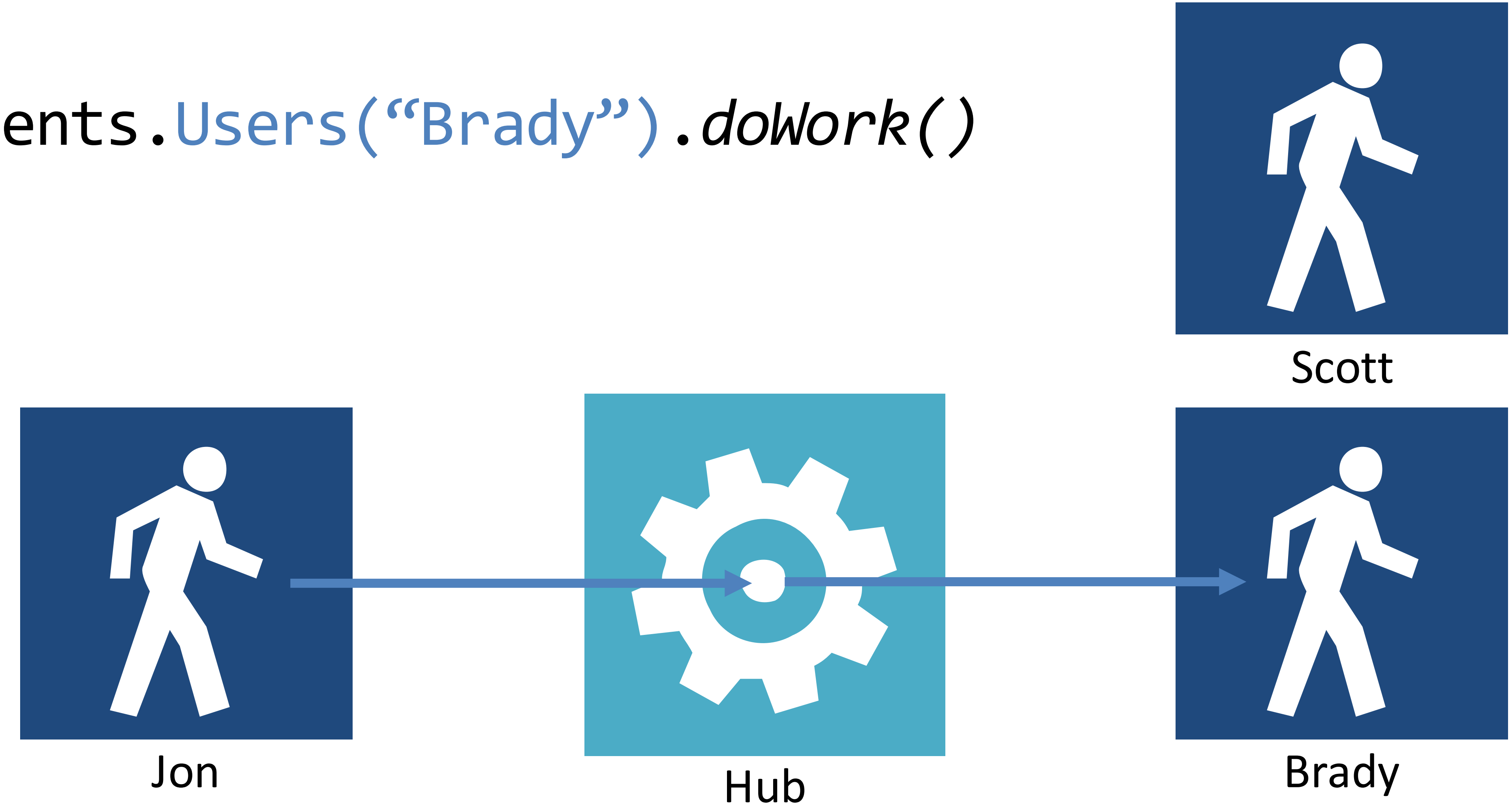
Sending messages to other connected clients

```
Clients.Others.doWork()
```



Sending messages to specific users

```
Clients.Users("Brady").doWork()
```



DEMO

Damian’s MoveShape Demo

MVA Microsoft Virtual Academy

Next Module – SignalR on the Web

Microsoft

Microsoft