SMS Texting with ColdFusion & AWS

by Greg Stanley

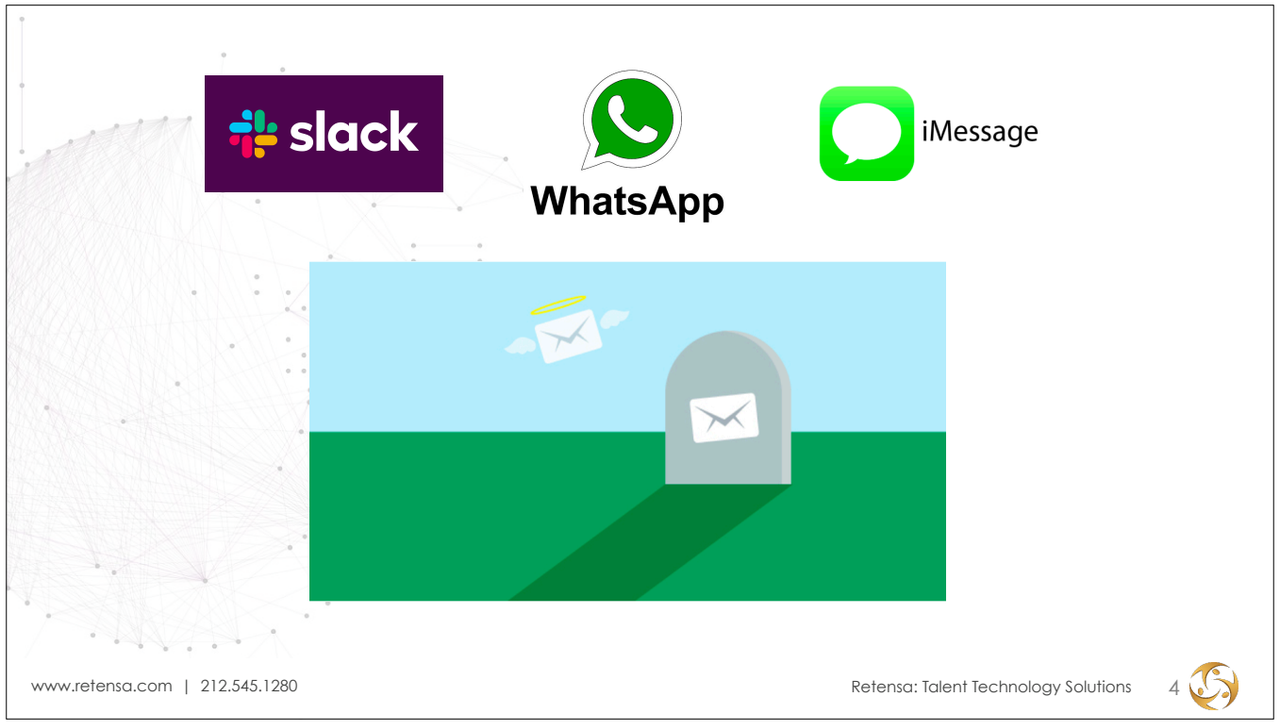
Intro and rundown of presentation.

We all know that “**companies** are only as **good** as their employees”

Retensa helps them **retain** their **best** employees.

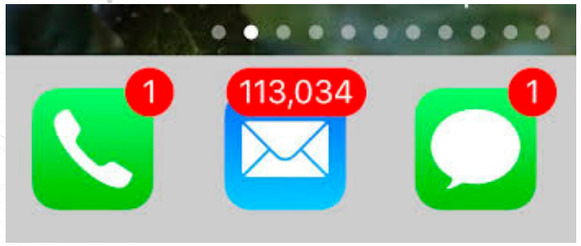
TalentPulse is our software, which I'll be showing you in a few minutes,
but one of the many things it does is deliver surveys to employees.

One of the challenges we have is to make sure these surveys get delivered to the people we send them to.



Raise your hand if you use chat applications?
Today, chat applications like **Slack**, **WhatsApp**, and **iMessage**, have all made emails somewhat obsolete.
So the best way to reach people nowadays, is with a text.

Think about it —



How many **unanswered** text messages to you have on your phone right now?
How many unanswered **emails** do you have?
Raise you hand - if you have more unanswered emails?
Receiving a survey in a **Text** is definitely more convenient for a lot of people, so we made it a priority in our software.

Today, I'm excited to share with you our SMS Texting solution, which uses **ColdFusion** and **Amazon Pinpoint**.

Capture the Employee Experience

TalentPulse

Automated
feedback system

- Automatic sending
- Advanced survey launcher
- Powerful reminder practice

Centralized
survey reporting

- Advanced dashboard
- Real-time insights
- Multi-rater friendly

Integrated
employee experience

- Tailored reports
- Custom survey design
- Customizable workflow

Data-driven change to develop, motivate, and retain top talent.

Our software is called **TalentPulse** — and the people that use it are HR and Talent Management depts.
Raise your hand — if you've ever taken an employee survey?
Ever wonder what happens to all of your answers?

Last generation of survey software

Most HR depts WANT to do something with them — but the last generation of survey software kind of sucks.

Capture the Employee Experience

TalentPulse

Automated
feedback system

- Automatic sending
- Advanced survey launcher
- Powerful reminder practice

Centralized
survey reporting

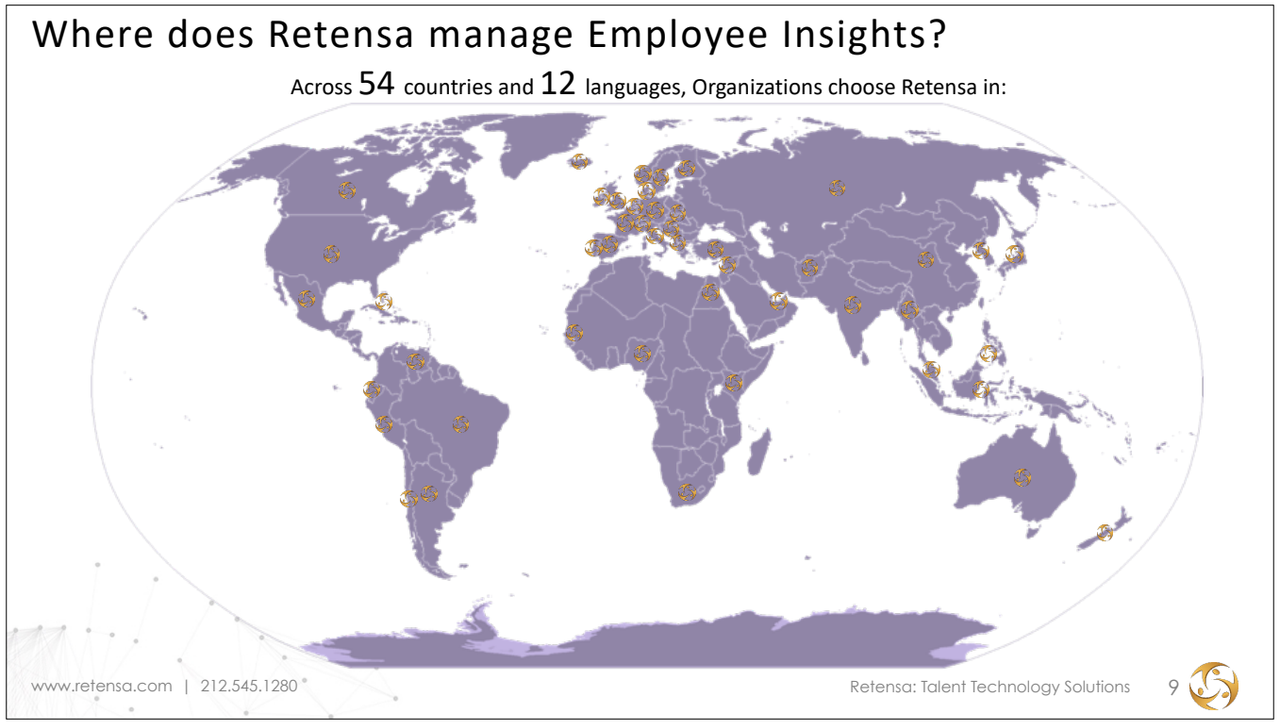
- Advanced dashboard
- Real-time insights
- Multi-rater friendly

Integrated
employee experience

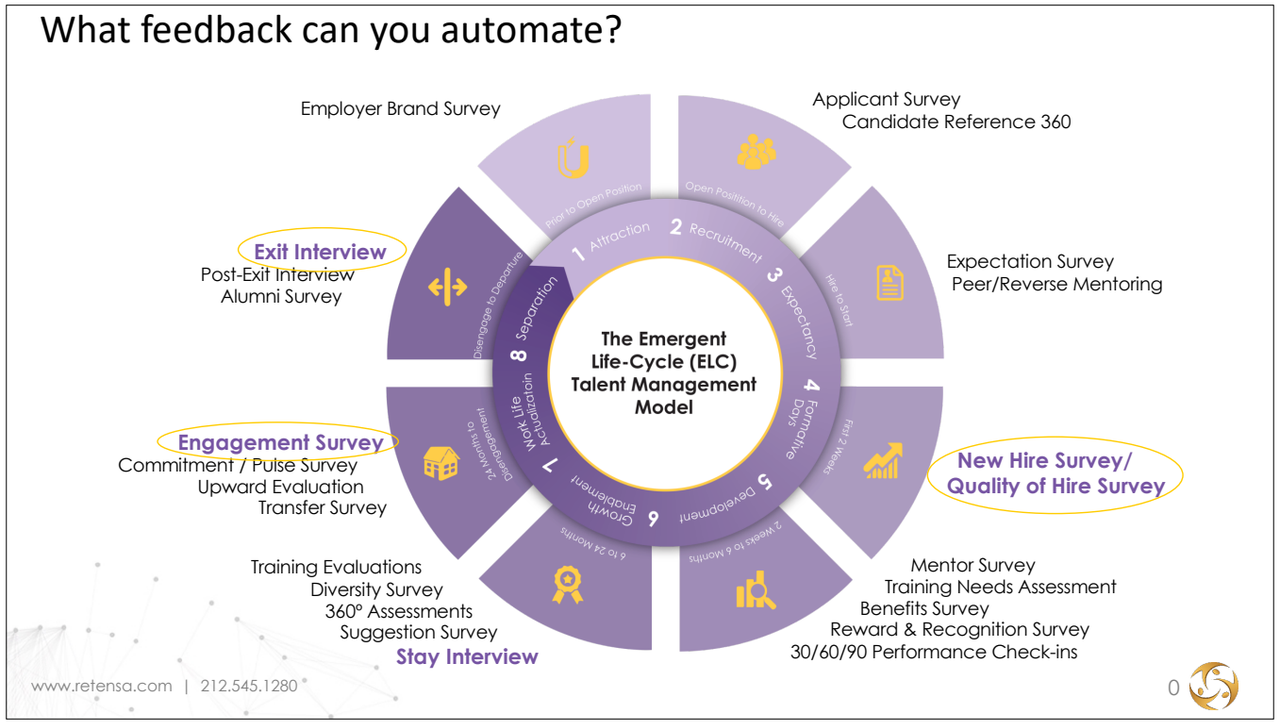
- Tailored reports
- Custom survey design
- Customizable workflow

Data-driven change to develop, motivate, and retain top talent.

So we built a platform that automates the collection of feedback — and streamlines the reporting of it.
So now instead of HR depts spending weeks struggling to make sense of the data — using Excel or whatever — They can take action faster and easier. And translate your answers into meaningful change at your company.



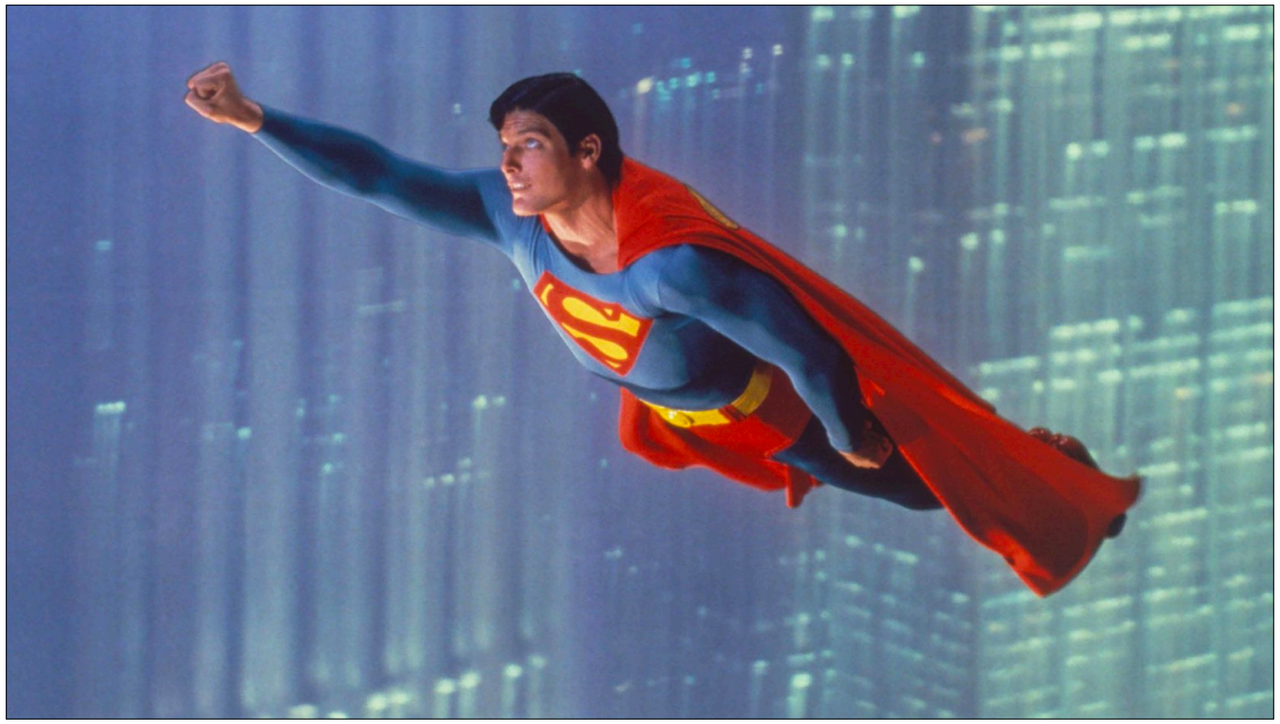
We currently have clients in 54 countries and in 12 languages.



TalentPulse **manages** the **entire** Employee Life Cycle — from **New Hire** Surveys — to **Exit Interviews** — and everything in between.

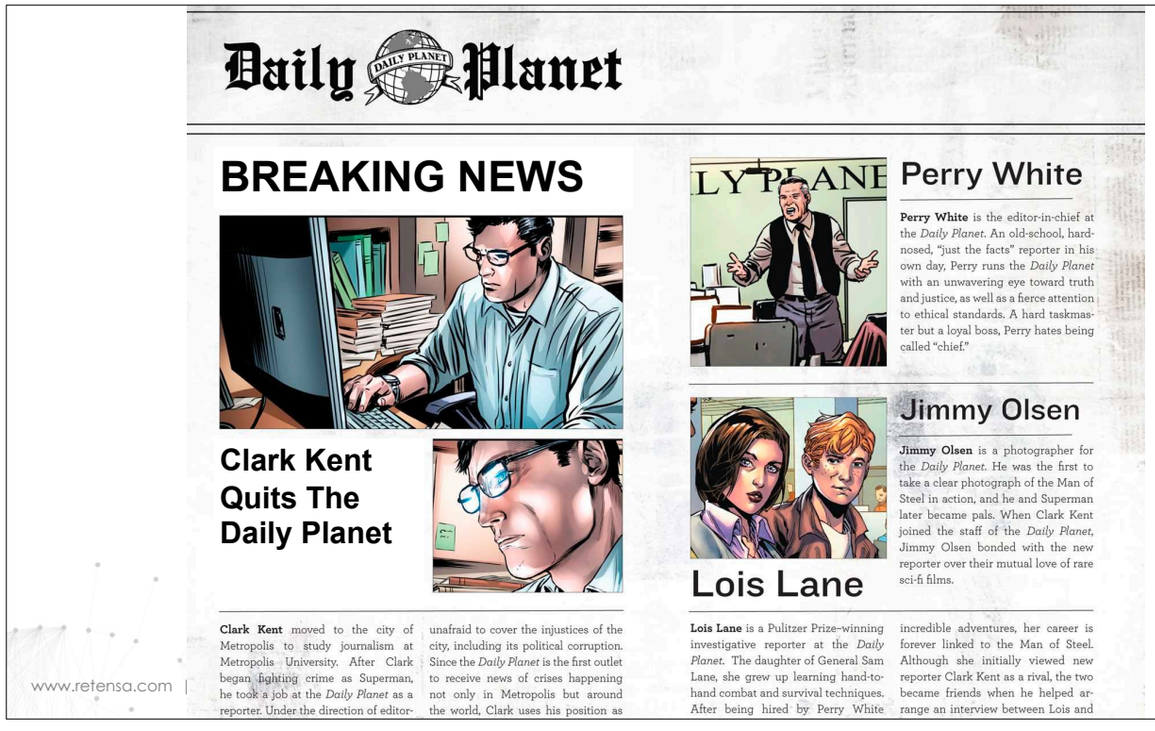
Demo Retensa's SMS Texting Solution

Raise your hand — if you're a fan of Superman - comics or movies?



Today, our fictional demo (of SMS Texting) takes place in the DC universe at The Daily Planet.
I don't know if you know this, but The Daily Planet has been a client of Retensa for some time now.
They use TalentPulse **everyday** for their employees.

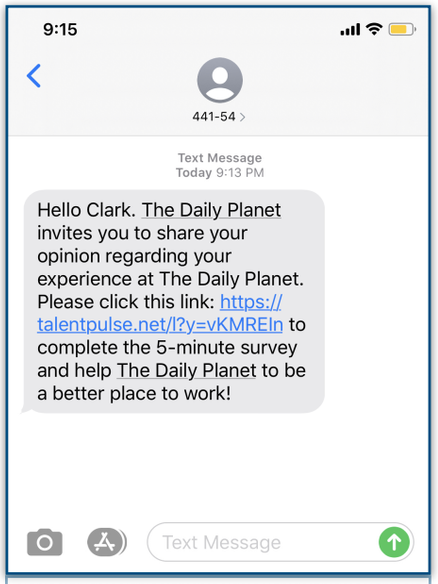
Well **today** is a big **day**, because it **happens** to be Clark Kent's last day at the Planet.



I **just got authorization** from HR at the Planet — and they're going to let us use TalentPulse to send Clark his Exit Interview Survey. Very cool. So let's do it!

EXIT SLIDES

SMS Text from Retensa

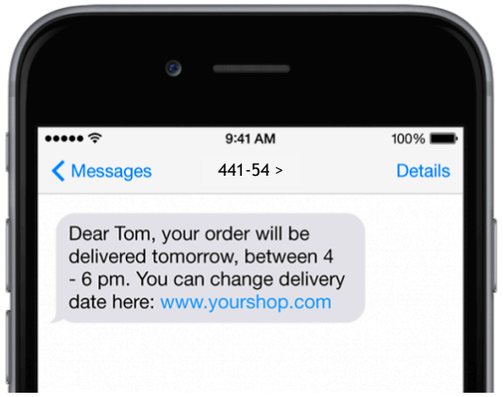


What the heck is Amazon Pinpoint?

What the heck is Amazon Pinpoint?

aws.amazon.com/pinpoint/features

- ☑ **SMS: Send text messages to your customers around the world**

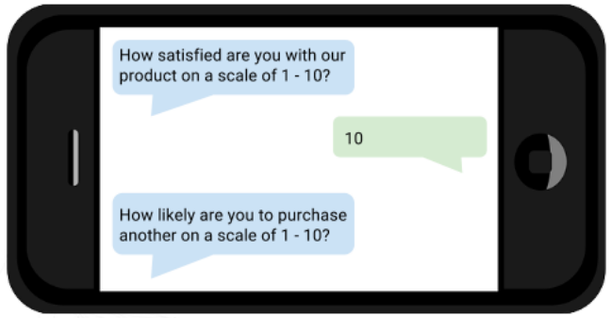


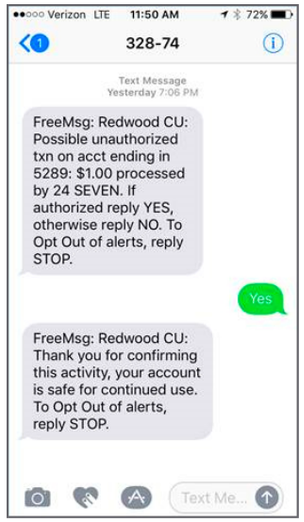
THIS presentation is all about sending SMS text messages… but there are other cool things Pinpoint can do.
One way text uses: airline check-ins, school reminders, package tracking status.

What the heck is Amazon Pinpoint?

aws.amazon.com/pinpoint/features

- SMS: Send text messages to your customers around the world
- **SMS: Create meaningful, interactive engagements (chat-like interactive experience)**





examples:
T-Mobile survey
Bank verifying charge

What the heck is Amazon Pinpoint?

aws.amazon.com/pinpoint/features

- SMS: Send text messages to your customers around the world
- SMS: Create meaningful, interactive engagements (chat-like interactive experience)
- **Push Notifications for mobile apps**
- **Email: Track email responses and other sending statistics (funnels - numbers of sends, deliveries, opens, clicks, bounces, complaints, and rejections)**
- **Voice: Programmatically make phone calls to your customers around the world. Turn your text into lifelike speech (Amazon Polly).**

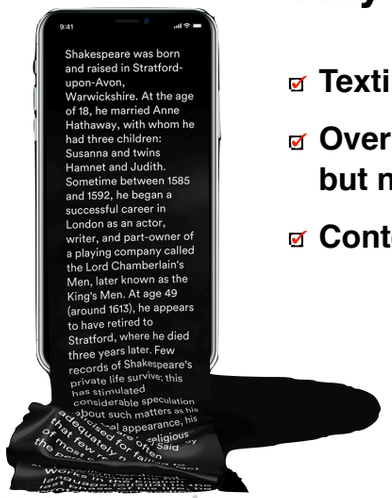
Why we chose Amazon Pinpoint for Sending Texts

- Can **send** text messages to US & International numbers
  docs.aws.amazon.com/pinpoint/latest/userguide/channels-sms-countries.html
- Can **validate** numbers to ensure they are mobile before sending (not available in SNS)
- **Opting out** is handled automatically (user can reply STOP to opt out of future texts)
- **Costs** $.00645 per request to **validate** any phone number
- **Costs** $.00645 per request to **send** in the US. A lot of the other countries are around that price too, however they vary and can be seen here: aws.amazon.com/pinpoint/pricing

We have clients in 54 countries - so we needed a robust solution that could send texts to the world, not just the US.

Talk about each!

Any Limits or Restrictions?



- Texting character limit
- Over 200 countries supported, but not all
- Content Restriction

Texting character limit? Although text messages are limited to 160 characters, most modern phones and networks support concatenation and segment and rebuild messages up to 1600 characters.

Over 200 countries - NOT Cuba, Iran, North Korea, Syria, and Sudan.

Content Restriction?
Most countries have restrictions on content
- ***Pornographic content***
- ***Content that has profanity or is hateful***
- ***Content that depicts or endorses violence***
- ***Content that endorses illegal drugs***

May be fees involved if it's reported

SMS Now or Later?

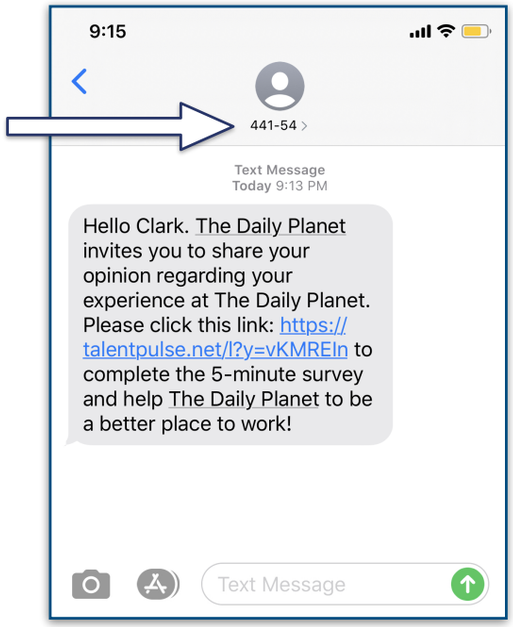
Message Types

- **Transactional** - Critical messages that support customer transactions, such as one-time passcodes for multi-factor authentication.
- **Promotional** - Noncritical messages, such as marketing messages.

Who sent that text?

Originating Sender (a few options):

- Free **Shared Short Code** (default)



We use free shared short code for one way texting.

Who sent that text?

Originating Sender (a few options):

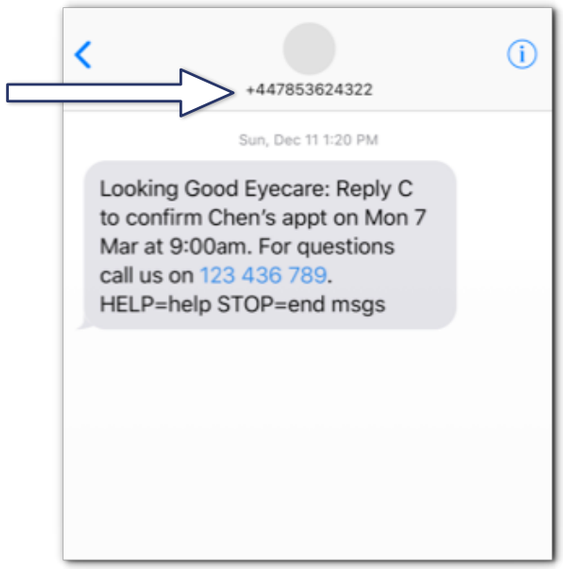
- Free Shared Short Code (default)
- **Dedicated Short Code** (pricey)

- **Dedicated Short Code**
- Needed if you want people to text you!!
- Takes 8-12 weeks to get one
- $650 setup fee
- $900 monthly fee.

Who sent that text?

Originating Sender (a few options):

- Free Shared Short Code (default)
- Dedicated Short Code (pricey)
- **Dedicated Long Code**

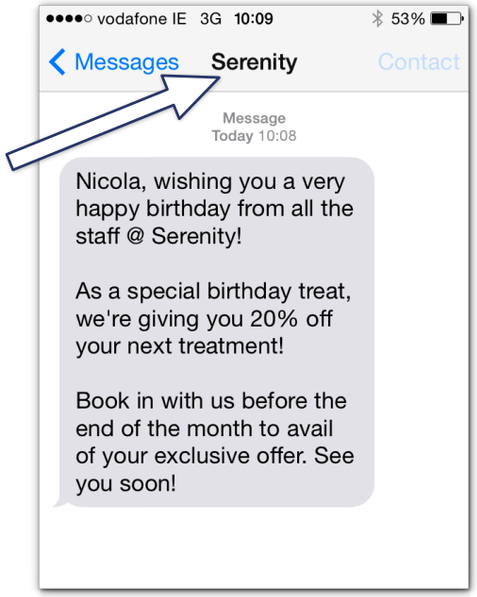


-**Long Codes**: 11 or 12 digit number (includes country code).
Amazon doesn't recommend using this if you are sending one way text messages.
Must request a long code from Amazon.
Limit 1 message per second.
Some other restrictions by country.

Who sent that text?

Originating Sender (a few options):

- Free Shared Short Code (default)
- Dedicated Short Code (pricey)
- Dedicated Long Code
- **Sender ID**

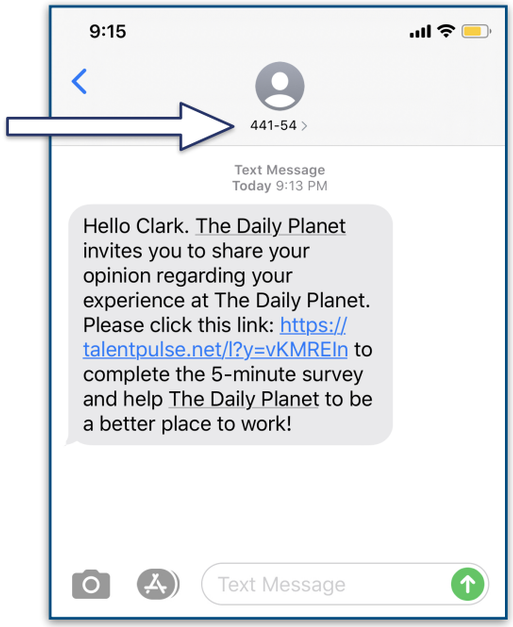


-**Sender IDs**: A brand or business name. Not supported in US, Canada and China.

Who sent that text?

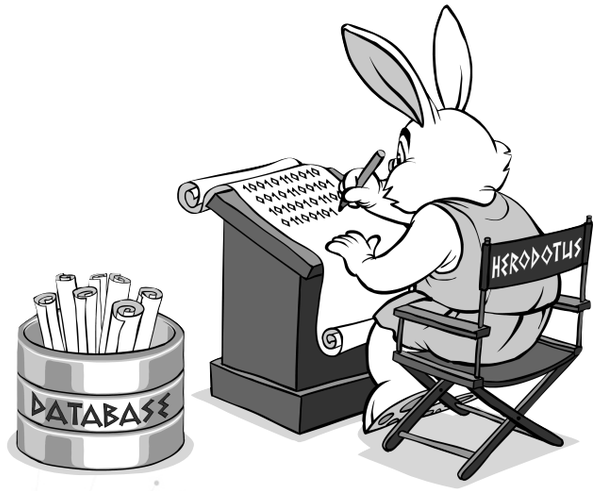
Originating Sender (a few options):

- Free **Shared Short Code** (default)



FOR ONE WAY TEXTING - We use free shared short code.

How To Minimize AWS Costs

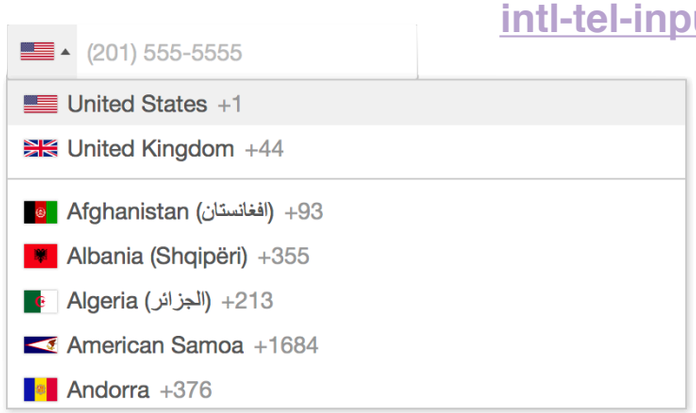


- ☑ Store validation results in the Database

So we never validate the same number more than once.

Create a Killer UX

☑ Use International Telephone Input
intl-tel-input.com



- free number formatting and digit validating for all countries
- cool flags
- Does NOT determine if mobile or landline

This is sweet!

How do I start using it?

What do you have to do to have full access to AWS services inside ColdFusion?

AWS SDK for Java

Download here: aws.amazon.com/sdk-for-java

Get started quickly using AWS with the AWS SDK for Java. The SDK helps take the complexity out of coding by providing Java APIs for AWS services including Amazon S3, Amazon ECS, DynamoDB, AWS Lambda, and more. The single, downloadable package includes the AWS Java library, code samples, and documentation.

Getting Started »

Developer Guide 2.0 »

API Documentation 2.0 »

Developer Blog »

Developer Guide 1.11 »

API Documentation 1.11 »

Downloads

AWS SDK for Java v2.0 »

Get the source on GitHub »

AWS SDK for Java v1.11 »

Get the source on GitHub »

AWS Toolkit for Eclipse »

Get the source on GitHub »

@awsforjava

You're going to need to add the Java SDK into ColdFusion.

AWS SDK for Java

Download here: aws.amazon.com/sdk-for-java

To have access to the full power of the AWS Java SDK, copy these .jar files to your cfusion/lib directory.

Adobe ColdFusion 2018:

- **aws-java-sdk-1.11.xxx.jar**

Adobe ColdFusion 2016 or 11:

- **aws-java-sdk-1.11.xxx.jar**
- jackson-annotations-2.6.0.jar
- jackson-core-2.6.7.jar
- jackson-databind-2.6.7.1.jar
- joda-time-2.8.1.jar

1.11 is still preferred method - since Adobe is starting to include that in CF.
Brian's awsPlaybox is also using v1.11.
CF 2020 will include AWS Java SDK!

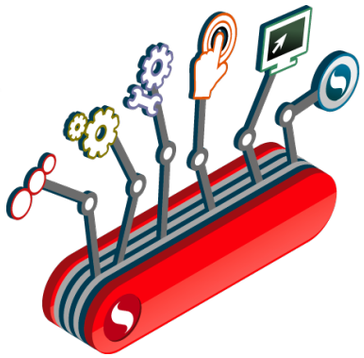
awsPlaybox

github.com/brianklaas/awsPlaybox

With the **awsPlaybox**, you'll be up and running with **ColdFusion** and **AWS** within a few minutes.

Currently, the **awsPlaybox** includes the following **AWS** modules:

- **DynamoDB**
- **IAM**
- **Lambda**
- **Rekognition**
- **SNS**
- **Step Functions**
- **Transcribe**
- **Translate**



Created by **Brian Klaas**

brianklaas.net

Big shout out to Brian!

What does the code look like?

model/awsCredentials.cfc - set your AWS keys

```
component accessors="true" output="false" hint="Simpler holder for AWS
credentials." {

   property name="accessKey" type="string";
   property name="secretKey" type="string";

   this.accessKey = 'YOUR IAM ACCESS KEY GOES HERE';
   this.secretKey = 'YOUR IAM SECRET KEY GOES HERE';

   /**
   *  @description Component initialization
   */
   public any function init() {
      return this;
   }
}
```

model/awsServiceFactory.cfc - add pinpoint code below

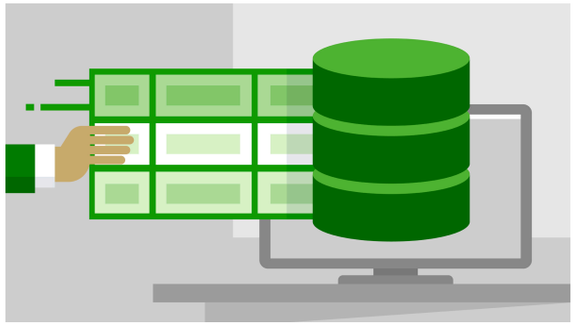
```
case 'rekognition':
   javaObjectName =
"com.amazonaws.services.rekognition.AmazonRekognitionClientBuilder";
   break;
case 'pinpoint':
   javaObjectName =
"com.amazonaws.services.pinpoint.AmazonPinpointClientBuilder";
   break;
case 'sns':
   javaObjectName = "com.amazonaws.services.sns.AmazonSNSClientBuilder";
   break;
```

application.cfc - add pinpoint code below

```
this.name = 'awsPlaybox';
this.applicationTimeout = CreateTimeSpan(0, 0, 15, 0);
this.sessionManagement = false;
this.datasource = 'my-cf-datasource';   /* added for Pinpoint */

.
.
.

// Pinpoint Project ID -- this is the Pinpoint Project ID in your AWS
account, under Pinpoint -> All Projects
application.pinpointProjectID = "1234567890";   /* added for Pinpoint
*/

return true;
```

How do I create the database tables used to minimize AWS costs?

- In the new **/pinpoint** directory, there is a file called **pinpoint.sql**, which has the script to create the 3 tables used in the code. Run this in your database.



Validate phone number using AWS Pinpoint

```
<cfset local.NumberValidateRequest =
   CreateObject('java',
   'com.amazonaws.services.pinpoint.model.NumberValidateRequest')
   .withPhoneNumber(arguments.phone)>

<cfset local.PhoneNumberValidateRequest =
   CreateObject('java',
   'com.amazonaws.services.pinpoint.model.PhoneNumberValidateRequest')
   .withNumberValidateRequest(local.NumberValidateRequest)>

<cfset local.pinpoint =
   application.awsServiceFactory.createServiceObject('pinpoint')>

<cfset local.pinres =
   local.pinpoint.phoneNumberValidate(local.PhoneNumberValidateRequest)>
```

Send Text Message using AWS Pinpoint

```
<cfset local.smsMessage =
    CreateObject('java', 'com.amazonaws.services.pinpoint.model.SMSMessage')
    .withBody(arguments.messageBody)
    .withMessageType('TRANSACTIONAL')>

<cfset local.directMessageConfiguration =
    CreateObject('java', 'com.amazonaws.services.pinpoint.model.DirectMessageConfiguration')
    .withSMSMessage(local.smsMessage)>

<cfset local.messageRequest =
    CreateObject('java', 'com.amazonaws.services.pinpoint.model.MessageRequest')
    .withMessageConfiguration(local.directMessageConfiguration)>

<cfset local.addressConfiguration =
    CreateObject('java', 'com.amazonaws.services.pinpoint.model.AddressConfiguration')
    .withBodyOverride(arguments.messageBody)
    .withChannelType('SMS')>

<cfset local.messageRequest.addAddressesEntry(arguments.phone, local.addressConfiguration)>
<cfset local.sendMessagesRequest =
    CreateObject('java', 'com.amazonaws.services.pinpoint.model.SendMessagesRequest')
    .withApplicationId(application.pinpointProjectID)
    .withMessageRequest(local.messageRequest)>
<cfset local.pinpoint = application.awsServiceFactory.createServiceObject('pinpoint') />
<cfset local.pinres = local.pinpoint.sendMessages(local.sendMessagesRequest)>
```

Some wrappers to make things easier

- **awsValidatePhone** - Validate a number with Pinpoint. Results are stored in table SMSValidateLog.
- **awsSendSMS** - Send a text message with Pinpoint. Results are stored in table SMSSendLog.
- **SMSCheck** - Checks the DB to see if the number was already validated, otherwise, **awsValidatePhone** is called and DB is updated with results.

Finish up with some wrappers to make life a little easier.
Pinpoint will not likely be included in CF 2020.

THANK YOU!

Discover the next generation of Talent Management solutions:

Employee Feedback Platform
www.TalentPulse.com

Exit Interview and Separation Management
www.TalentPulse.com

Employee Retention Strategies
www.Retensa.com

Want 10 Free Exit Interviews for your company?
Visit retensa.com/event42

Greg Stanley
gregstanley1@gmail.com
Twitter: **gregstanley1**

Presentation Slides and **Demo Code** will be available for **download** here:
gregstanley.com/pinpoint