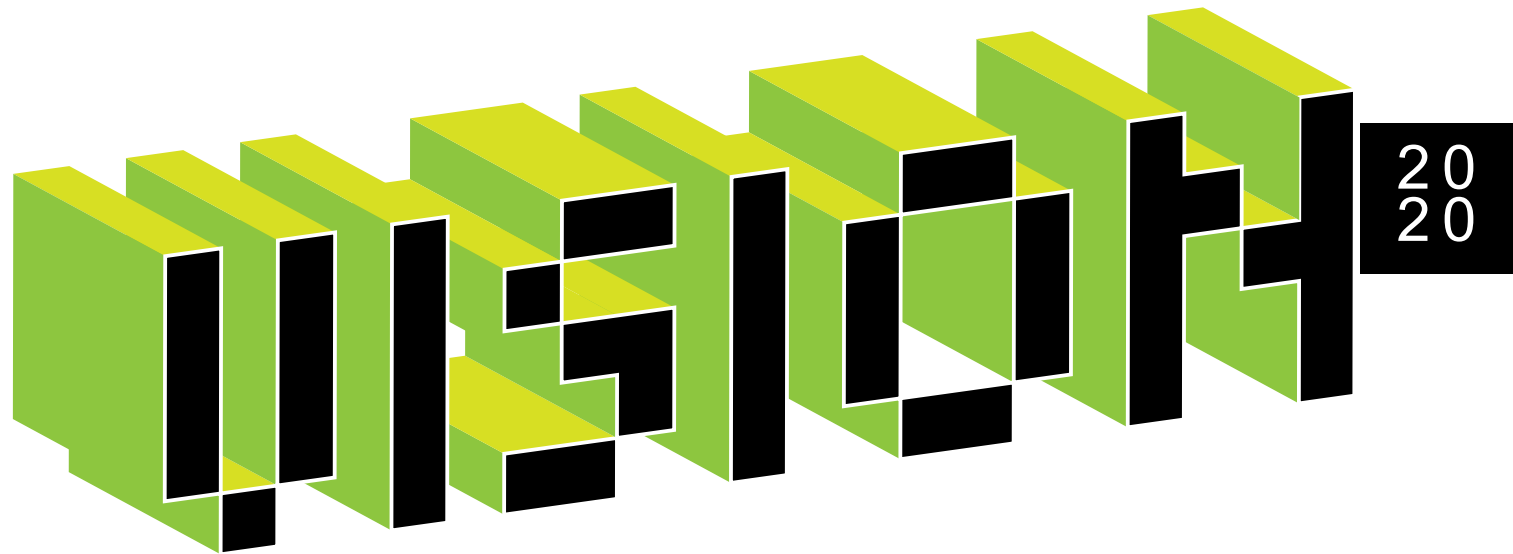


**JADE**





**W E L C O M E**

JADE USER GROUP 2020

# Virtual JUG: Vision 2020



Time	Topic	Speakers
9:00 am	<b>Intro and Business Updates</b>	<i>John Ascroft, Charlotte Walshe, Henry Varney</i>
9:25 am	<b>REST easy: client, server, security</b>	<i>Tim Aitchison</i>
10:05 am	<b>A quick break</b>	
10:10 am	<b>JADE 2020 Roadmap &amp; feature demos</b>	<i>Iryna Melnyk, John Richards, Hugh McColl, William Cameron, Hugh Mercer-Beumelburg, Luke Derry</i>
10:50 am	<b>Containerisation demo</b>	<i>Hugh McColl</i>
11:10 am	<b>A quick break</b>	
11:15 am	<b>BeeJay's IDE Tips and Tricks</b>	<i>BeeJay Johnstone</i>



**Charlotte Walshe**

Jade Business Update

# 2019 The Year That Was



- Small profit, JADE revenues strong, re-focussing digital
- Doubled sales team size globally:
  - 2 salespeople in UK,
  - 4 in Auck,
  - 1 in Chch,
  - 2 in Aus
- Strong growth in Jade ThirdEye in UK
- Invested in product management, new tools for collaboration, and learning and development for our people

# 2020 Full Steam Ahead



- Continued focus on helping our customers grow with the support of a strong Sales team
- Expanding Jade ThirdEye into Australia
- JADE 2020 release
- Covid19 – Jade very fortunate, 2/3rd recurring revenue, very supportive shareholder and Board, great creative people, and customers who want us to keep helping them to grow
- Q&A



**Henry Varney**

Skipton Business Update



2019 Annual  
Results Update

16 July 2020



# Business and Financial Highlights



## **1,046,294 members**

Society member numbers increased by 35,868 to 1,046,294 (2018: by 91,366)



## **£1.2bn Savings balances growth**

Savings balances grew by £1.2bn to £17.4bn by the end of 2019, an annual growth rate of 7.6% (2018: by £1.1bn, a growth rate of 7.7%)



## **£4.8bn Gross residential mortgage lending**

Gross residential mortgage lending was up 13.1% at £4.9bn (2018: £4.3bn) despite intensive competition in the mortgage market. Group net interest margin was 0.98% (2018: 1.09%)



## **£1.9bn Mortgage book growth**

The mortgage book grew by £1.9bn to £20.0bn, a growth rate of 10.3% (2018: £1.6bn, a growth rate of 10.0%) with Group UK residential mortgage arrears by three months or more representing only 0.25% of mortgage accounts (2018: 0.30%), a 16% year on year reduction



## **£153.2m Total Group Profit Before Tax (PBT)**

Total Group PBT was £153.2m, below the level of 2018 (2018: £188.7m) due to the reduction in underlying profits and also includes fair value losses of £3.4m (2018: gain of £1.0m) relating to the legacy equity release portfolio



## **£155.2m Underlying Group PBT**

Underlying Group PBT was £155.2m, representing 4.1% of Risk Weighted Assets (2018: £186.6m, 4.4% of RWA's)



## **39.1% Group Common Equity Tier 1 ratio**

The Common Equity Tier 1 ratio strengthened significantly to 39.1% (2018: 32.8%) and the Leverage ratio remained strong at 6.0% (2018: 6.1%)



## **9.9% Increase in Group Total Assets**

Group Total Assets increased 9.9% during the year to £25.5bn (2018: 23.2bn)

*“Further improvements in asset quality, maintained capital strength & strong profitability”*



## **COVID-19 Impact**

*“Further improvements in asset quality, maintained capital strength & strong profitability”*

# JADE Stats



## **240,000 Members registered online**

Over 240,000 members using our Customer Portal website.



## **130,000 Members registered to use our Mobile App**

Launched in 2019 we have 130,000 members using our mobile app.



## **11 Million digital transactions**

Our digital channels generated over 11 million JADE transactions in February using our API interface.



## **96% Completed < 0.5sec**

41 million JADE transactions processed in February - 96% < 0.5 second - 99% < 1 second.



## **850gb Database**

In the last 12 months we have seen an increase of 100gb in our JADE database.



## **27TB for JADE Databases**

Over 27TB of disk taken up by 32 replicas for development, testing and resiliency purposes.

*“Further improvements in asset quality, maintained capital strength & strong profitability”*



### **Running JADE 2018**

All systems were upgraded from Jade 7.1 to J2018 in May 2019.

Favourite features include

- type methods,
- parametrised constructors and
- IDE performance and customisation improvements.



**Timothy Aitchison**

REST Easy

# JADE REST Client

# RESTful Web Services

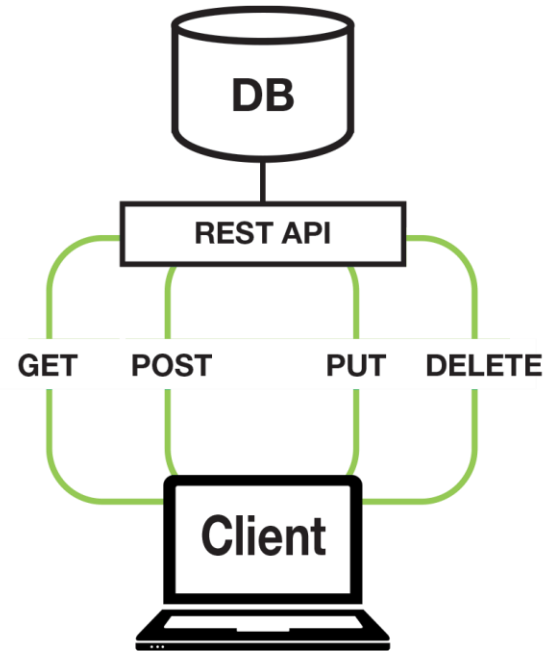


REST Server Side

**JADE** ✓

REST Client Side

**JADE** ?





# JADE 2020



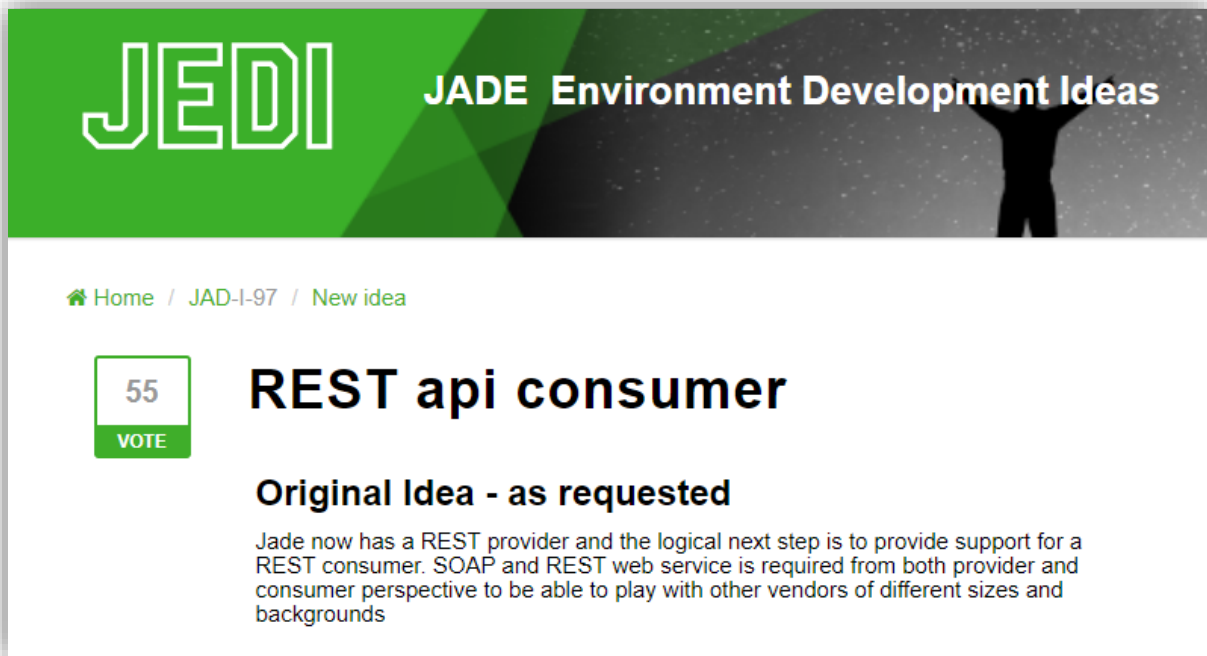
## REST Server

- Improvements in JADE 2020

## REST Client

- New in JADE 2020

# JADE 2020

A screenshot of a web page for a JEDI (JADE Environment Development Idea) titled "REST api consumer". The page has a green header with the "JEDI" logo and the text "JADE Environment Development Ideas". Below the header is a navigation breadcrumb: "Home / JAD-I-97 / New idea". The main content area features a vote count of 55 in a green box labeled "VOTE", followed by the title "REST api consumer" and the subtitle "Original Idea - as requested". The description text reads: "Jade now has a REST provider and the logical next step is to provide support for a REST consumer. SOAP and REST web service is required from both provider and consumer perspective to be able to play with other vendors of different sizes and backgrounds".

**JEDI** JADE Environment Development Ideas

[Home](#) / [JAD-I-97](#) / [New idea](#)

**55**  
VOTE

## REST api consumer

### Original Idea - as requested

Jade now has a REST provider and the logical next step is to provide support for a REST consumer. SOAP and REST web service is required from both provider and consumer perspective to be able to play with other vendors of different sizes and backgrounds

# The 3 classes ...

JadeRestClient
endpoint
get post delete put

**JadeRestClient**  
Responsible for sending requests and generating responses.

JadeRestRequest
body header uri
addObjectParam addURLSeg

**JadeRestRequest**  
Responsible for generating the appropriate URI, headers and body.

JadeRestResponse
data lastErrorCode responseStatus
deserialize

**JadeRestResponse**  
Responsible for holding response information and deserializing response objects.

# RESTful Web Services



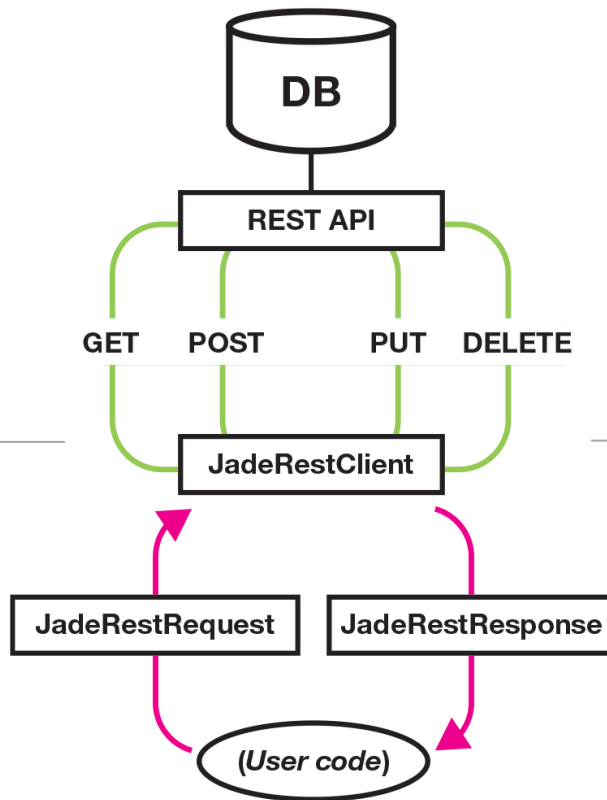
REST Server Side

JADE



REST Client Side

JADE



# REST in JADE



*So, let's make **REST** really easy...*

# Introducing ...



Open API  
Specification



Swagger

} **3.0**

# What is the Open API Specification?



A standard for documenting REST APIs



Allows for auto-generation of proxy classes



Human and machine readable



Includes the data model of the API

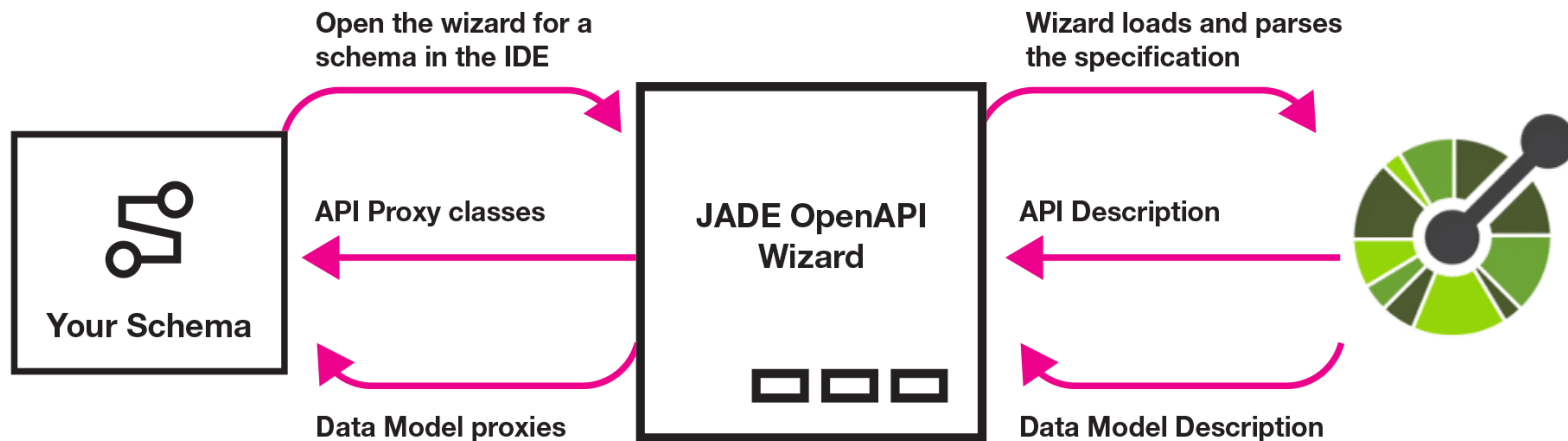


Obtained from server with simple REST request



Saves developers time and effort

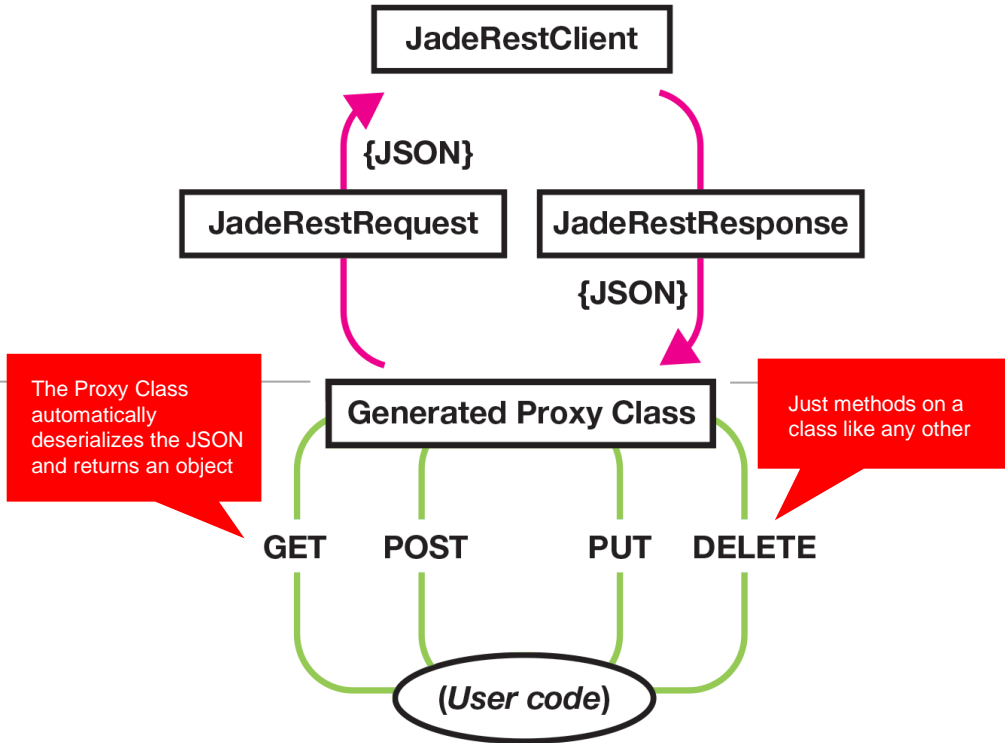
# Using OpenAPI with JADE





# Use either layer

## JADE REST Client



## Proxy Classes

# REST in JADE



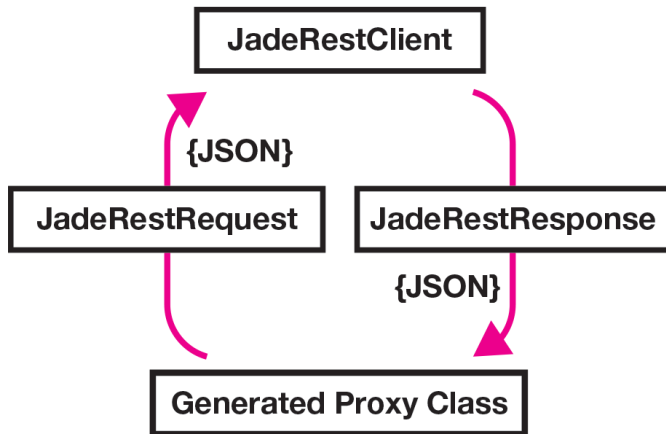
*What if I want to use the proxy classes,  
but still have low-level control?*

# JADE REST Client Hooks



# JADE REST Client Hooks

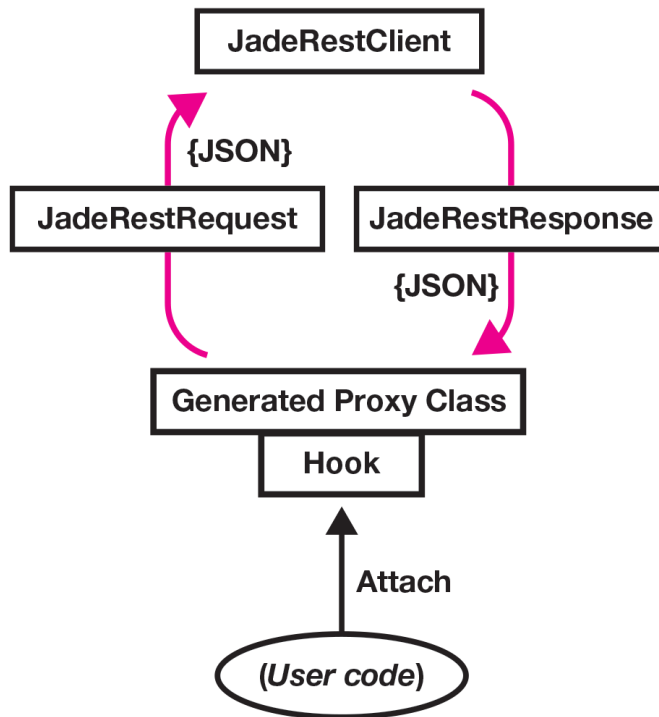
JadeRestProxyHook



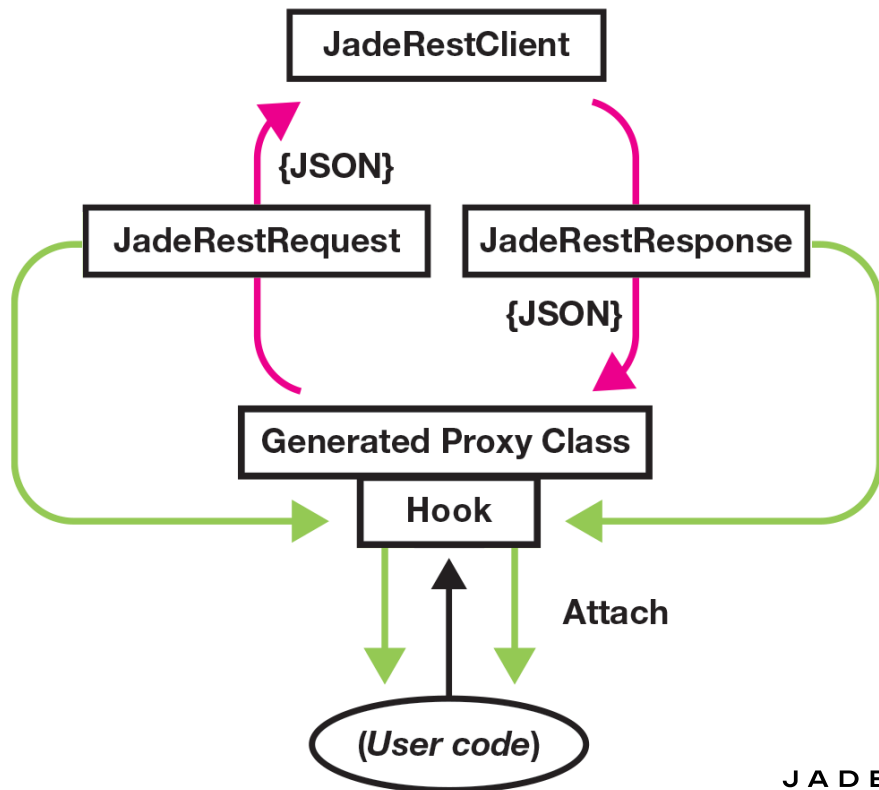
- *preSerializeHook()*
- *preSendHook()*
- *postSendHook()*

(User code)

# JADE REST Client Hooks



# JADE REST Client Hooks



# REST in JADE



*Let's see it in **action** ...*



# Demo Time!

Here's what to look out for:



## ASP.NET API

Introduction to the demo/testing API  
(written in ASP.NET)



## Loading the spec

Using the JADE  
OpenAPI Import  
Wizard.



## What we get

Let's see the API  
Proxy Classes and  
Data Model



## Using the API

Easy to use!



# JADE 2020

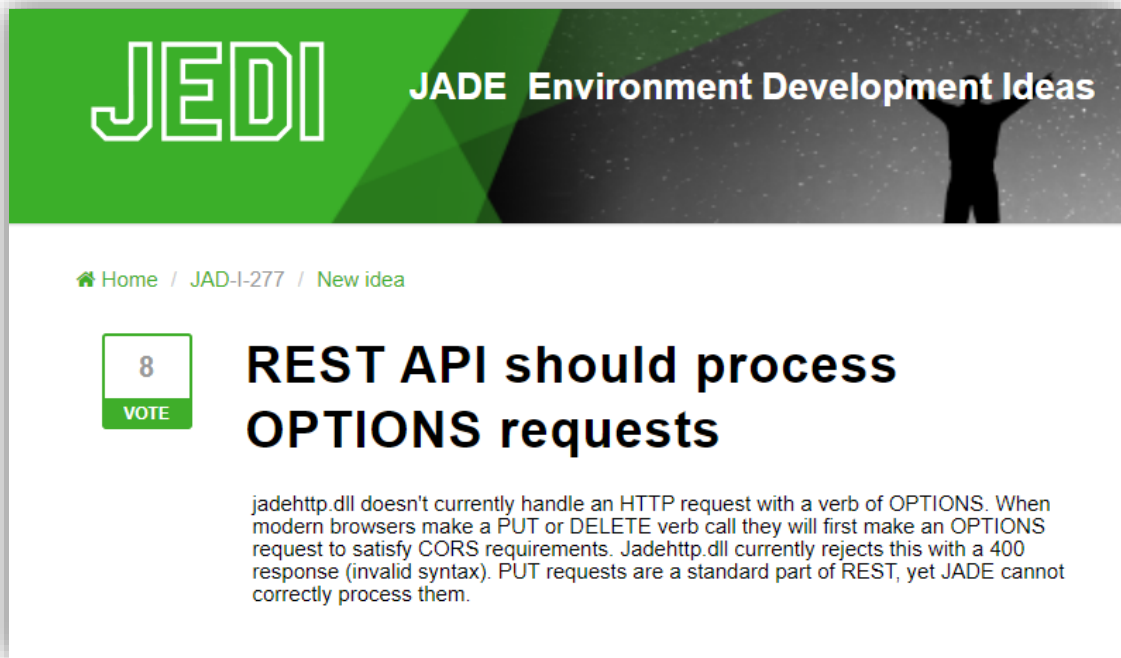


## REST Server

- Improvements in JADE 2020

## REST Client

- New in JADE 2020
- Makes consuming REST easy
- Integrates with OpenAPI

A screenshot of a web page titled "JADE Environment Development Ideas". The page has a green header with the "JEDI" logo and the title. Below the header is a navigation breadcrumb: "Home / JAD-I-277 / New idea". The main content area features a vote counter showing "8" and a "VOTE" button. The title of the idea is "REST API should process OPTIONS requests". The description explains that jadehttp.dll does not currently handle OPTIONS requests, which are used by modern browsers for CORS requirements, and that jadehttp.dll currently rejects such requests with a 400 error.

## JEDI JADE Environment Development Ideas

[Home](#) / [JAD-I-277](#) / [New idea](#)

**8**  
VOTE

### REST API should process OPTIONS requests

jadehttp.dll doesn't currently handle an HTTP request with a verb of OPTIONS. When modern browsers make a PUT or DELETE verb call they will first make an OPTIONS request to satisfy CORS requirements. Jadehttp.dll currently rejects this with a 400 response (invalid syntax). PUT requests are a standard part of REST, yet JADE cannot correctly process them.

A banner for JEDI (JADE Environment Development Ideas) with a green background on the left and a dark, starry background on the right. The word "JEDI" is in large white letters, and "JADE Environment Development Ideas" is in smaller white letters. A silhouette of a person with arms raised is visible on the right side.

[Home](#) / [JAD-I-430](#) / [New idea](#)

1

VOTE

## Securing RESTful services

Currently, JADE provides RESTful web services with the `JadeRestService RootSchema` class. However, these APIs are not able to be secured, that is, anyone with the URL may consume it without providing authentication.

# JADE 2020



Header

```
base64enc({
  "alg": "HS256",
  "typ": "JWT"
})
```

Payload

```
base64enc({
  "iss": "toptal.com",
  "exp": 1426420800,
  "company": "Toptal",
  "awesome": true
})
```

Signature

```
HMACSHA256(
  base64enc(header)
  + '.' +
  base64enc(payload)
  , secretKey)
```



# JADE 2020



```
exampleJWT();

vars
  jwt      : JadeJsonWebToken;
  claim    : JadeJWTClaim;
  exp      : TimeStamp;
  date     : Date;
  time     : Time;
begin
  create jwt;
  jwt.addIssuer("JADE");

  date.setDate(1,12,2020); // First of December 2020
  time.setTime(17,0,0,0); // 5pm
  exp.setDate(date);
  exp.setTime(time);
  jwt.addExpiry(exp);

  claim := create JadeJWTClaim("accessLevel", "admin");
  jwt.addCustomClaim(claim);
  write jwt.encodeHS512("Some secret password, only known by the server");
end;
```



**eyJhbGciOiJIUzUxMiIsInR5cCI6IkpXVCJ9.  
eyJpYXQiOiJlOTQyMTEzNzQsImFjY2VzcyI6ImVlbnVlcnRtaW4iLCJpc3MiOiJKQURFliwiZXhwIjoxNjA2ODQyMDAwfQ.**

**pBtuzSiRUz9I8P\_Cm7ciRVL0yMdFOvalpN  
Ack2tIlZbSM1\_if-  
FkVTg3GD1ianWDK36zuFI8YvQG0UA8T6A  
U5A**

# JADE 2020



```
exampleOldWay();

constants
  Endpoint = "http://localhost/RestfulJade/jadehttp.dll";
  Resource = "Customer";
  AppName = "RestService";

vars
  http : JadeHTTPConnection;

begin
  create http transient;

  http.url := Endpoint & "/" & Resource & "?" & AppName;

  http.sendRequest(http.Verb_GET, "Authorization: Bearer " & getJWT(), null);

  write http.readBody(0).String;
end;
```

```
exampleRestClient();

vars
  client      : JadeRestClient;
  request     : JadeRestRequest;
  response    : JadeRestResponse;

begin
  client := create JadeRestClient("http://localhost/RestfulJade/jadehttp.dll");

  request := create JadeRestRequest("Customer");

  // The token may be from the REST server itself, or from a 3rd party.
  request.addBearerToken(self.getJWT()); // self.getJWT is a placeholder

  response := create JadeRestResponse();

  client.get(request, response);
end;
```

# JADE 2020



The screenshot shows the "MyRestSchema Class Browser" for the "MyRestService" class. The left pane displays a tree view with "MyRestService" selected. The middle pane shows the class signature "getCustomer():" and the start of its implementation with "vars", "begin", and "end:". The right pane shows a context menu with "REST Security Options..." highlighted. Below the editor, it says "Source synchronized".

The "AddClaimForm" dialog box is shown. It has a title bar with a "J" icon and standard window controls. The "Method:" field is set to "PROTOTYPE". Below it, "Select Claim Type:" is a dropdown menu with "Single-Value Claim" selected, and an "Add Claim" button. The "Added Claims:" section contains a list with "userName must be admin" selected, and "Edit" and "Remove" buttons. At the bottom, there is a "WORK IN PROGRESS" label and "OK" and "Help" buttons.

# JADE 2020



## REST Server

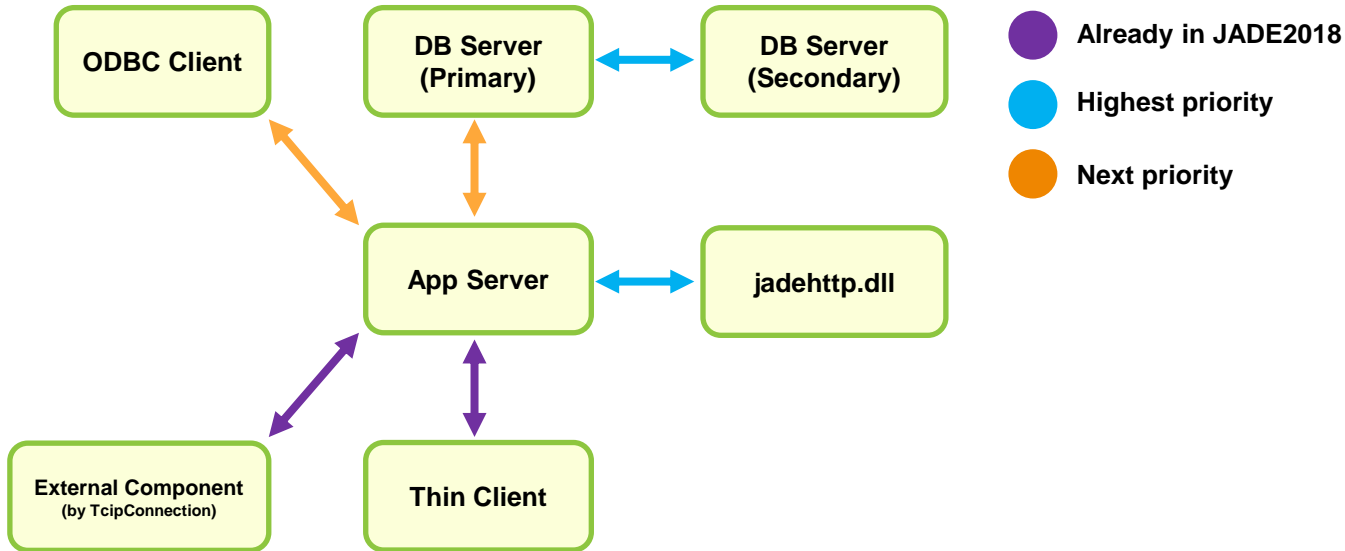
- Improvements in JADE 2020
- More HTTP Verbs supported
- Enhanced Security with JWTs

## REST Client

- New in JADE 2020
- Makes consuming REST easy
- Integrates with OpenAPI



# JADE 2020



# Looking Forward...



**JADE 2020**

- REST Server improvements
- REST Client implementation
- Lots of other features
- What next?

REST = EASY



JADE



# **JADE 2020 Roadmap**

Iryna Melnyk, John Richards, Hugh McColl, William  
Cameron, Hugh Mercer-Beumelburg, Luke Derry



# What we'll talk about

- JADE Release schedule
- 2020 Roadmap
- Feature demos
- Additional announcements
- Q&A



# Vision

JADE provides the most effective way for our customers to meet their **evolving** development and runtime needs, through a **reliable, high performance, easily managed** platform that integrates with other technologies and platforms.

# Vision

## How did we plan the 2020 roadmap?



**Listen**  
to our customers



**Pay attention**  
to trends



**Invest**  
and invigorate

# JADE Release schedule





# Live roadmap



<https://jedi.ideas.jadeworld.com>

- Ready for Release
- Executing
- Planned (2020)

A screenshot of the JEDI IDE Environment Development Ideas website. The header features the 'JEDI' logo and the text 'JADE Environment Development Ideas'. Below the header, there's a navigation bar with 'Add a new idea', 'Recent', 'Trending', and 'Popular' tabs, along with a search bar. The main content area displays a list of ideas, each with a vote count, a title, a description, and a status label. The ideas shown are: 'Source control overhaul' (45 votes, Future consideration), 'Null conditional operator' (41 votes, Future consideration), 'JADE SDS traffic encryption' (38 votes, Executing), and 'Right-click should select' (28 votes, IDE Backlog). A sidebar on the left provides a filter by status and category, with 'Planned (2020)' highlighted in green.

**JEDI** JADE Environment Development Ideas

[Add a new idea](#) Recent Trending Popular

**Filter by Status**

Needs review	63
Future consideration (Long term)	90
Future consideration: 2022	26
IDE Backlog	76
Planned (2020)	3
Executing	17
Ready for Release	68
Shipped	64
Closed	43
Customer Feedback Requested	7

**Filter by Category**

Admin	0
DevOps	24
IDE & debugger	60
Interoperability	133
Language and Compiler	13
Root Schema	28
Runtime	19
User Interface & Skins	16

[Log in / Sign up](#)

**45 VOTE** **Source control overhaul**  
For a few years now there have been discussions on how Jade could be improved by including a new and modern take on source control, utilising industry accepted tools or ideas. A vote here is firstly then to confirm you still feel this is important...  
Created about 1 year ago by John Deaufoy DevOps Future consideration (Long term) 7

**41 VOTE** **Null conditional operator**  
Null exceptions must be up there with the most common runtime issue. Often these occur when dealing with older persistent data that didn't have the same validation enforced on it, or object references yet to be populated. There are various strateg...  
Created about 1 year ago by John Deaufoy Language and Compiler Future consideration (Long term) 0

**38 VOTE** **JADE SDS traffic encryption**  
Not all SDS traffic goes over a private link or VPN. We would like SDS traffic to have the ability to be encrypted if required.  
Created about 1 year ago by Guest DevOps Executing 0

**28 VOTE** **Right-click should select**  
It would be nice if the Jade IDE worked the same as the rest of Windows does. If I right-click an entry in a list, it should select that entry, and present the context menu for it. Jade works differently from the rest of Windows in that only left...  
Created about 1 year ago by Guest IDE & debugger IDE Backlog 0

# JADE 2020 Roadmap



**Openness**



**DevOps**



**User Experience**



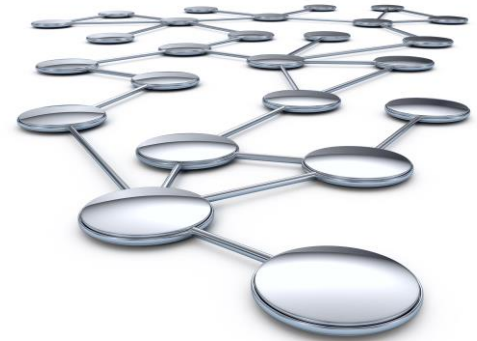
**Performance**



**Developer Experience**



**Security**





# Openness



- Support Utf8 in JadeHttpConnection ([JAD-I-349](#))
- REST API Consumer ([JAD-I-97](#))
  - JADE REST client RootSchema classes
  - GET, PUT, POST and DELETE
  - OPTIONS requests

[Home](#) / [JAD-I-97](#) / [New idea](#)

56

VOTE

## REST api consumer

**Original Idea - as requested**

Jade now has a REST provider and the logical next step is to provide support for a REST consumer. SOAP and REST web service is required from both provider and consumer perspective to be able to play with other vendors of different sizes and backgrounds

### Feature Summary

Currently, JADE is able to provide RESTful web services using the JadeRestService RootSchema class, but it is not able to easily consume them. Limited functionality is

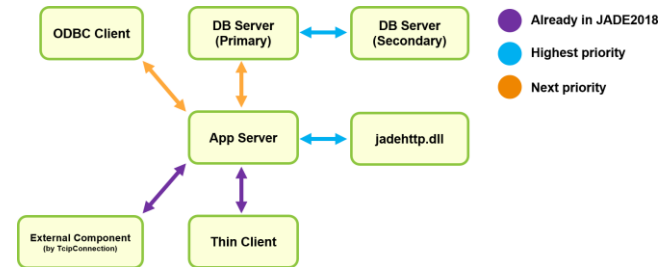
Total: **2** ready for release



# Advanced Security



- Securing RESTful services ([JAD-I-430](#))
  - JSON Web Tokens (JWT)
- End to End SSL Protection ([JAD-I-431](#))
  - High priority group
    1. jadehttp.dll <> app server
    2. primary <> secondary  
([JADE SDS traffic encryption JAD-I-275](#))
  - Medium priority group
    1. app server <> rap
    2. jadewebsockets.dll <> app server
  - Lower priority group
    1. odbc thin client
    2. JADE to JADE webservices



Total: **3** executing



# User Experience



- Skins – Usability improvements:
  - Skin Category Selection in Painter ([JAD-I-404](#))
  - JadeTableSkin to allow the definition for the fixed row and column fore color ([JAD-I-443](#))
  - Ability to change a table scrollbar skin ([JAD-I-306](#))
- Float/Dock for MDI Child forms ([JAD-I-446](#))
- Message Boxes with custom button captions ([JAD-I-146](#))
- comboBox enhancements for touch screens ([JAD-I-467](#))

Total: **1** executing, **5** ready for release



# Developer Productivity



- RootSchema Iterable / Iterator Interfaces ([JAD-I-429](#), [JAD-I-173](#)):
  - An abstraction above collection classes, enabling non-collection classes to be the target of the "foreach" statement.
  - A way to re-use generic code with different collection or non-collection "iterable" inputs.
- Array of variable length strings ([JAD-I-84](#))
- Regular Expression Support ([JAD-I-438](#))

Total: **2** executing, **1** ready for release



# Developer Productivity: Regex

with William Cameron



## Regular Expression Support ([JAD-I-438](#))

5

VOTE

### Regular Expression Support

OverviewThe proposal is to provide a Pattern Matching API be added to the JADE product. This would be provided as a set of JADE classes.ObjectiveA key objective of the proposal is:To make it easier for developers by reducing the amount of hand rol...



Created 10 months ago by Ashley Bass  
Language and Compiler

Ready for Release



0



# Developer Productivity: Regex



## Why use this feature?

Develop fast optimized string parsing and manipulation that can be quicker to read and less error prone than hand rolled alternatives.

### Hand rolled email validation

```
1 matchOld();
2 constants
3   Email = "me@jadeworld.com";
4   Email2 = "me@blahdeblah.com";
5 vars
6   result : Boolean;
7   c : Character;
8   atSymbolFound : Boolean;
9   posAtSymbolWasFound : Integer;
10  i : Integer;
11 begin
12   foreach i in 1 to Email.length do
13     c := Email[i];
14     if not c.isNumeric and not c.isAlpha and c <> '.' and c <> '-' then
15       if c = '@' and not atSymbolFound and i <> 1 then
16         // no @ at the start and only 1 of them
17         atSymbolFound := true;
18         posAtSymbolWasFound := i;
19         continue;
20       endif;
21       write "false";
22       return;
23     endif;
24   endforeach;
25
26   if atSymbolFound and posAtSymbolWasFound < i then
27     write true;
28   else
29     write false;
30   endif;
31 end;
```

Object locked

### Regex email validation

```
1 matchNew();
2
3 constants
4   Email = "me@jadeworld.com";
5   Email2 = "me@blahdeblah.com";
6 begin
7   write JadeRegex@isMatch(Email, "[\w.-]+@[\\w.-]+", true);
8 end;
9
```





# Developer Productivity: Regex



The screenshot shows the JADE IDE interface. The title bar reads "JADE [D:\systems\jade-regex-development\system - cnwvc5 [System User] : singleUser] - [RootSchema Class Browser: JadeRegexPattern]". The menu bar includes File, Edit, Jade, Options, Browse, View, Classes, Properties, Methods, History, Window, and Help. The toolbar contains various icons for navigation and editing. The left sidebar shows a class tree with "JadeRegexPattern" selected. The main window is divided into three panes: "All" (showing attributes like \_memAddress, nativeCompileOptionFlags, nativeExtraCompileOptionFlags, nativeMatchOptionFlags, optionFlags, patternString, and timeout), "Attributes" (empty), and "References" (empty). The right pane shows a list of methods, with "setAllowEmptyClass" selected. Below the panes, a code editor shows the following code:

```
isMatch(textToSearch : String) : Boolean is regexPatternIsMatch in jomsupp final;
```



# Developer Productivity: Regex



## Resources:

- [Regex101](#)
  - Has an in-depth reference
  - A visual regex debugger
  - Develop your regex before putting it in JADE
- [RegexOne](#)
  - Teaches you basic and advanced concepts via an interactive tutorial
  - The syntax is slightly different to ours, but it is still a useful resource for learning



# Developer Experience

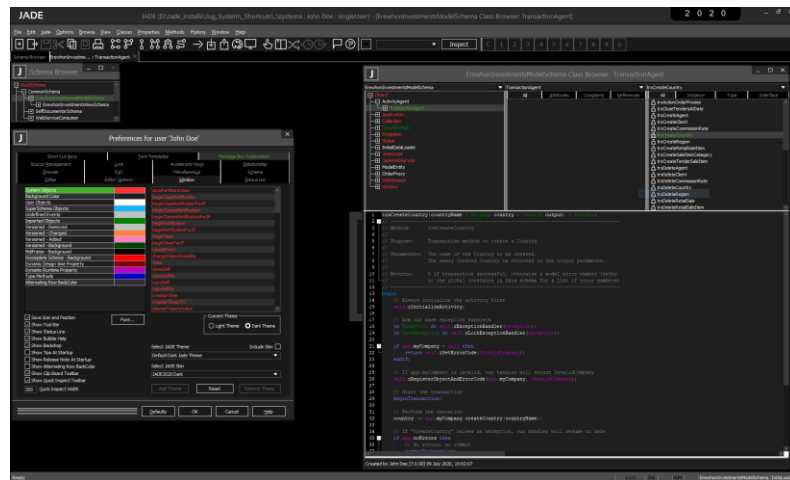
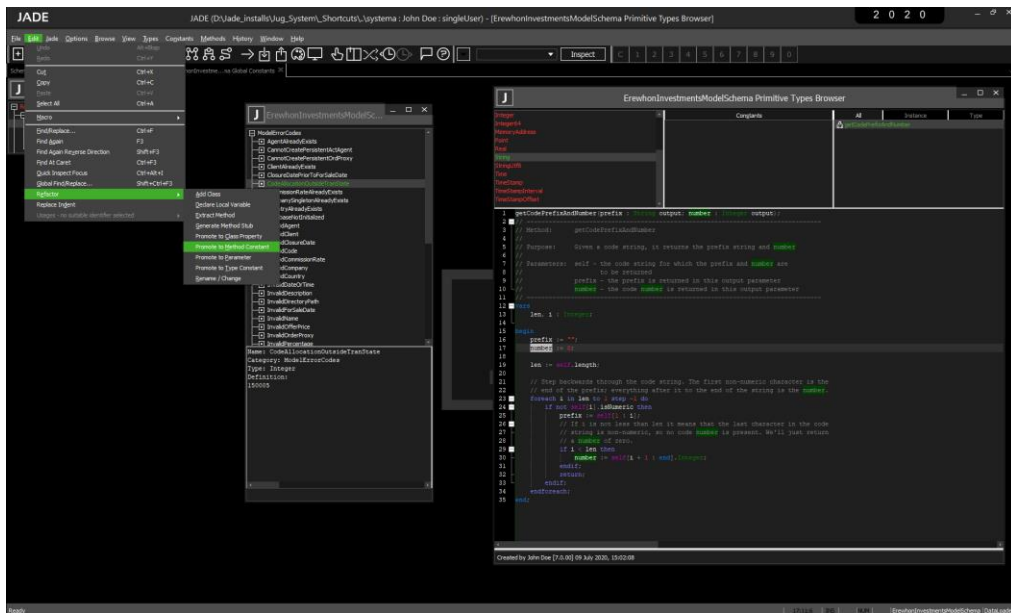


- Built-in Quick Inspect ([JAD-I-168](#))
- IDE UX improvements:
  - IDE Themes ([JAD-I-432](#))
    - Including Dark Themes ([JAD-I-370](#))
  - Coloured access icons ([JAD-I-348](#))
  - Tab colour scheme ([JAD-I-352](#))
- [Smaller improvements](#) to the IDE (40+ features, including UI, efficiency, layouts etc.)
- More small improvements could be added from [IDE Backlog](#)

Total: **1** planned, **6** executing, **48** ready for release



# Dark IDE Themes



18  
VOTE

## Allow for a dark IDE scheme

I use dark themes and schemes for all software that supports it as I find it easier on my eyes after prolonged use. Much software allows for this these days. It would be helpful if Jade both supported a dark theme and allowed for a dark theme to b...

Created about 1 year ago by Alister Melville  
IDE & debugger

Executing 4



# Built-in Quick inspect

with Luke Derry



The screenshot shows the JADF Schema Inspector for Client [2106.1]. The interface includes a menu bar (File, Edit, Options, Help), a Schema Browser on the left, and a main display area with three panes: History, Search, and Inspect. The Inspect pane shows the details of a selected client, including their name, address, and contact information.

Client

History Search

--- 1 to 20 of 20 ---

- Client [2106.1]
- Client [2106.2]
- Client [2106.3]
- Client [2106.4]
- Client [2106.5]
- Client [2106.6]
- Client [2106.7]
- Client [2106.8]
- Client [2106.9]
- Client [2106.10]
- Client [2106.11]
- Client [2106.12]
- Client [2106.13]
- Client [2106.14]
- Client [2106.15]
- Client [2106.16]
- Client [2106.17]
- Client [2106.18]
- Client [2106.19]
- Client [2106.20]

Class Number: 2106

Inspect

Client

Inspect

Schema Workspace X

```
---Client[2106.1---  
address1 = "24c Dryside Place"  
address2 = "Christchurch"  
address3 = "New Zealand"  
email = "wild@more.co.nz"  
fax = "64 3 923 1010"  
name = "Pauline Wild"  
phone = "64 3 834 7172"  
webSite = <null>  
allRetailSales = RetailSaleByTimeItemDict/2139.1.2106.4.4  
allTenderSales = TenderSaleByTimeItemDict/2247.1.2106.4  
allTenders = TenderByItemTimeOfferDict/2144.1.2106.4.1  
myCompany = Company/2107.1.1
```

Inspecting multiple objects

EreshonInvestmentsModelSchema Data.Loader



# JADE Time zones

*with Hugh Mercer-Beumelburg*



- Support Time zones and daylight savings ([JAD-I-254](#))

17

VOTE

## Converting a Timestamp to UTC and daylight savings issues

When we need to get a UTC timestamp (for example creating a calendar .ics file) we would use the following code:

```
vars
```



# JADE Time zones



The screenshot shows the JADE IDE interface. The main editor displays the following code:

```
1 example_meetingTime();
2
3 vars
4   clientTimeZone: JadeTimeZone;
5   serverTimeZone: JadeTimeZone;
6   meetingTime: Time;
7   meetingDate: Date;
8   meetingTimeStamp: TimeStamp;
9 begin
10  clientTimeZone := JadeTimeZone@createTimeZoneByLocationWindows(PresentationClient);
11  serverTimeZone := JadeTimeZone@createTimeZoneByLocationWindows(DatabaseServer);
12
13  //Set the meeting time
14  meetingTime.setTime(18,0,0,0); //The meeting is at 6pm
15  meetingDate.setDate(5,10,2020); //The meeting is on the 5th October 2020
16  meetingTimeStamp.setTime(meetingTime);
17  meetingTimeStamp.setDate(meetingDate);
18
19  write "The meeting is on " & meetingTimeStamp.String & " in " & clientTimeZone.displayName;
20  write "The meeting is on " & clientTimeZone.convertTimeByTimeZone(meetingTimeStamp, serverTimeZone).String
    & " in " & serverTimeZone.displayName;
21 epilog
22   delete clientTimeZone;
23   delete serverTimeZone;
```

The Jade Interpreter Output Viewer shows the following output:

```
The meeting is on 05 October 2020, 18:00:00 in (UTC+12:00) Auckland, Wellington
The meeting is on 05 October 2020, 06:00:00 in (UTC+00:00) Dublin, Edinburgh, Lisbon, London
```

The IDE status bar at the bottom indicates "Execution complete - elapsed time 00:00:00" and "Ready".



# DevOps



- Restartable logical certify ([JAD-I-478](#))
- Certifier warnings added ([JAD-I-407](#))
- Source Control ([JAD-I-477](#))
- Containerising JADE services ([JAD-I-433](#))

Total: **2** planned





# DevOps: Source Control

*with John Richards*



## Source Management improvements ([JAD-I-477](#))

- Improve Patch History Management by:
  - Make finding things easier and faster
  - Group changes better
  - Links to Change Management Tools
  - Make code comparisons easier and faster
- Areas to be addressed:
  - Improve performance
  - Associate a "ticket number" for tracking
  - Search efficiently by Patch No, Ticket no, Entity name, Developer, Date / Time, Status
  - Remove bugs: make it more reliable and predictable



# Performance

*with Hugh McColl*



## Collection Concurrency Enhancements ([JAD-I-423](#))

Make it easier for developers to write code that maintains persistent Collections either directly or as a side-effect of inverse maintenance in a manner that minimizes contention and avoids deadlocks.

### New RootSchema methods

- tryAddDeferred, tryRemoveDeferred, plus dictionary methods

### Deferred execution strategy for inverse maintenance

- Specify deferred execution when defining an inverse relationship
- Enable or disable with Process method



# Benchmark - Scenarios



## Interactive Workload

Work **10ms**

shareLock / unlock a collection

Work **10ms**

**beginTransaction;**

Update the collection

Work **10ms**

**commitTransaction;**

## Batch Workload

reserveLock **For non deferred - required to prevent deadlocks**

**beginTransaction;**

Update 100 objects with inverses **10ms work per batch**

**commitTransaction;**



# Benchmark - Results



Test Case	Non-Deferred Elapsed Time	Deferred Elapsed Time	Change
<b>Interactive:</b>			
- 5 users	127ms	72ms	44%
- 5 users, with no initial read (share lock) access	110ms	68ms	38%
- 5 users, collection updated at end of transaction	71ms	70ms	0.7%
- Single user	60ms	61ms	-0.6%
<b>Batch:</b>			
- 5 processes, 4 Collections updated	7.2 secs	2.3 secs	69%
- 5 processes, 3 Collections updated	5.6 secs	2.1 secs	62%

# JADE 2020 Roadmap



**Openness**



**DevOps**



**User Experience**



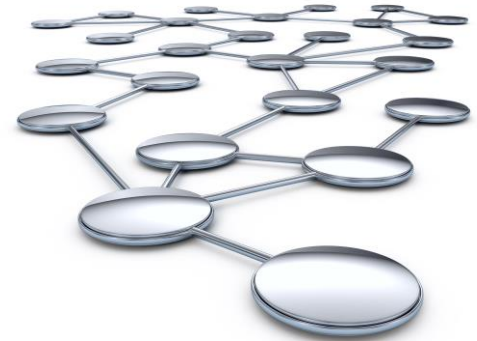
**Performance**



**Developer Experience**



**Security**



# JADE 2020: Coming in Q4



<https://www.jadeworld.com/developer-center>

Newsletter

E-mail announcement

# Future of JADE on the JEDI portal



<https://jedi.ideas.jadeworld.com>

Check “Future consideration: 2022” and give us your feedback!

A screenshot of the JEDI portal interface. The header features the "JEDI" logo and the text "JADE Environment Development Ideas" against a background of a person's silhouette against a cloudy sky. Below the header, there are navigation tabs for "Recent", "Trending", and "Popular", along with a search bar. On the left, there is a sidebar with a "Filter by Status" section (listing "My ideas", "My votes", "Needs review", "Future consideration (Long term)", "Future consideration: 2022", "IDE Backlog", "Planned (2020)", "Executing", "Ready for Release", "Shipped", "Closed", "Customer Feedback Requested") and a "Filter by Category" section (listing "Admin", "DevOps", "IDE &amp; debugger", "Interoperability", "Language and Compiler", "Root Schema", "Runtime", "User Interface &amp; Skins"). The main content area displays three idea cards. The first card, titled "SDS Zero RPO (Transaction Level Synchronicity)", has 3 votes and is categorized as "Future consideration: 2022". The second card, titled "Generator style iterator (yield return / yield break)", also has 3 votes and is categorized as "Future consideration: 2022". The third card, titled "JADE IDE Extensions", has 13 votes and is categorized as "Future consideration: 2022".



# Future of JADE on the JEDI portal

1. Generator style iterator (yield return/break) — **JAD-I-524**
2. Method Overloading — **JAD-I-425**
3. Primitive Type SubTyping — **JAD-I-290**
4. JADE IDE Extensions — **JAD-I-383**
5. Enumerations — **JAD-I-81, JAD-I-428**
6. RPS with Cloud specific SQL DBMS — **JAD-I-525**
7. File streams and an I/O Stream abstraction — **JAD-I-403**
8. Auto generate Responsive Web/Mobile user interfaces
9. SDS Zero Transaction loss capability (Zero RPO capability) — **JAD-I-526**
10. Coroutine Support — **JAD-I-292**



# Additional Announcements



## Developer training:

- Online Developer Course [registrations are Open](#)
- Extended topics ([JAD-I-488 for feedback](#))

JADE TOI for JADE 2020 to be announced

## Communication:

- Quarterly Newsletter
- JEDI Digest
- Slack community
- Facebook group

# Join Our Community-led Groups



Slack:



Facebook: Look up “JADE User Group”





**Hugh McColl**  
Containerising JADE



# Containerising JADE Services



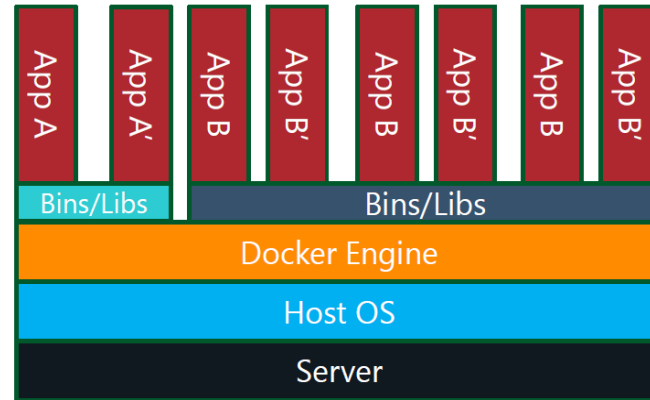
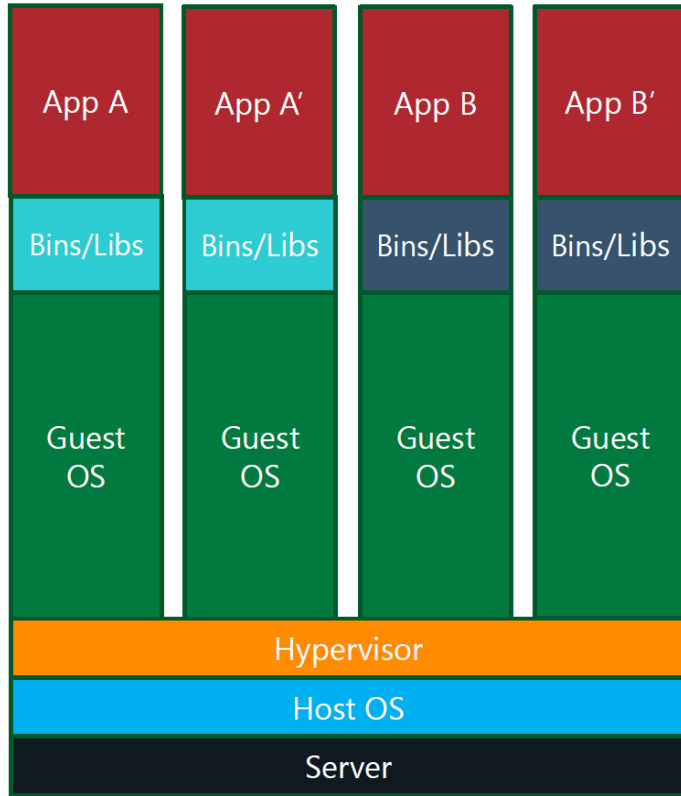
## Containerising JADE Services ([JAD-I-433](#))

- Containers - what are they?
- Container Benefits
- JADE 2020 deliverables
- Demo – JADE containers in action



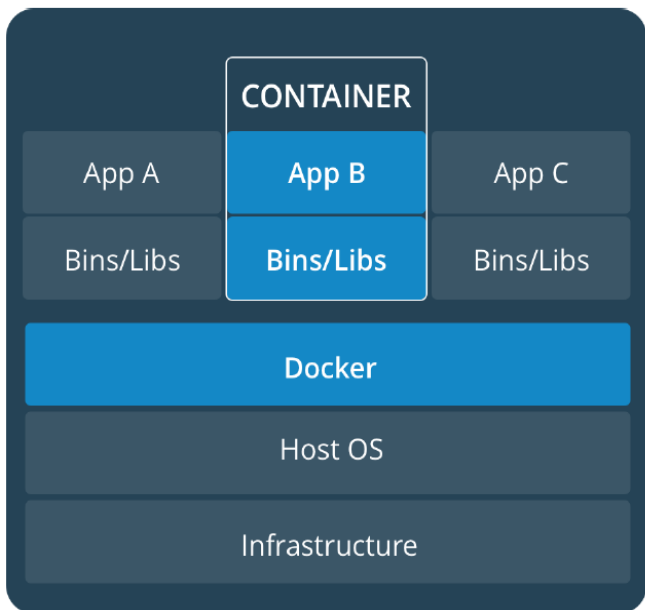
# Virtual machine vs container

Better consolidation ratio

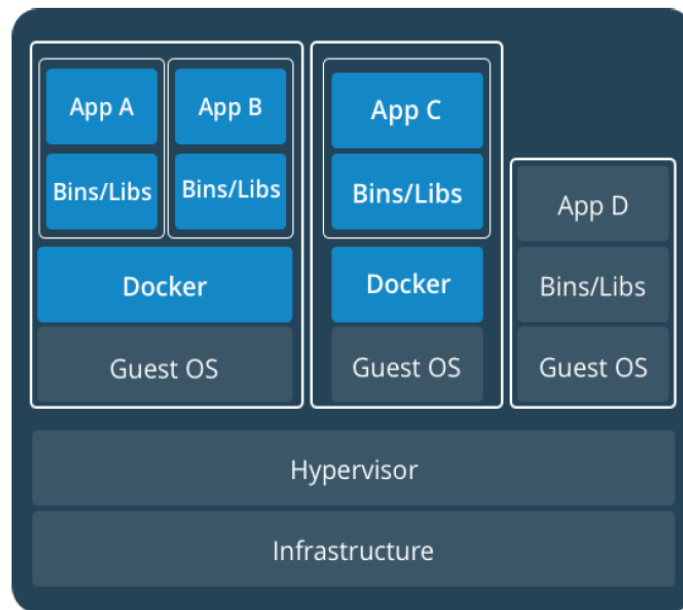


# Virtual Machine vs container

Can be run on bare metal

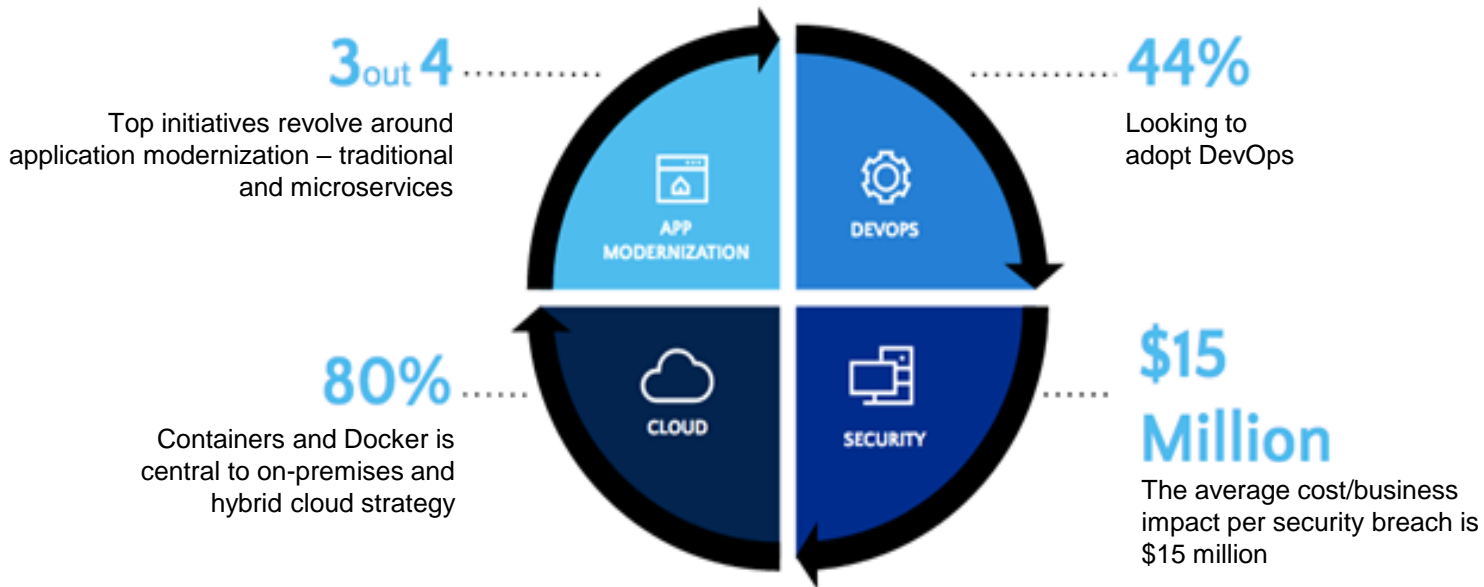


Can be run inside a VM



# Business Drivers

## Top drivers for customer application modernization



1. State of App Development Survey, Q1 2016 | Morgan Stanley, CIO Survey, June 30, 2016  
2. Source: Ponemon Institute, "2015 Cost of Cyber Crime - US"





# 2020 Deliverables

- Container aware hosting processes
  - jadrapb (new)
  - jadappb (existing)
- Removal of non-essential GUI dependencies from back-end components
- Base Images accessible from a public container registry
- Sample Dockerfile and deployment scripts published on the public JADE Github repository: <https://github.com/jadesoftwarenz>
- Logging solution

JEDI Idea: [JAD-I-433](#)





# Docker Concepts



**Dockerfile:** A build script



**Image:** A packaged service

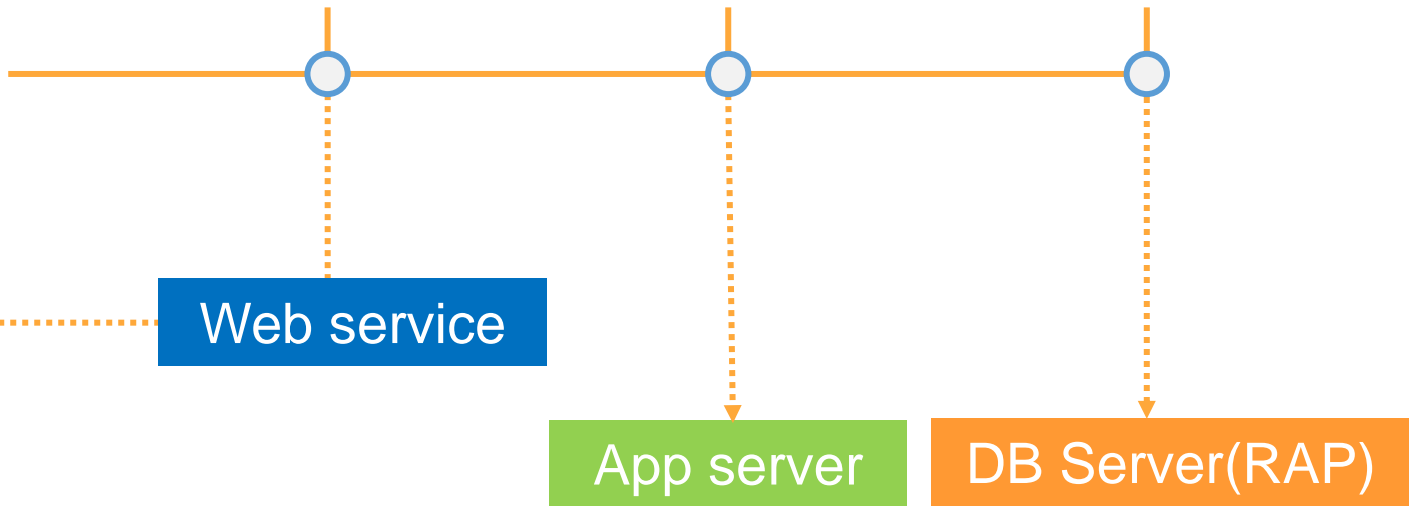
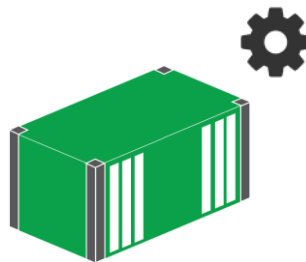
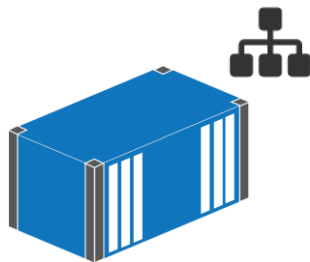
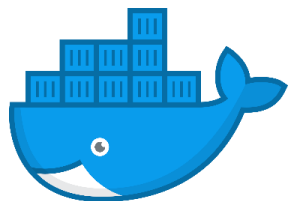


**Container:** A running service



**Registry:** An image store

# Deployment Scenario



# JADE Container Demo



*JADE Containers in action.*





# **BeeJay's IDE tips & tricks**

Brian (BJ) Johnstone



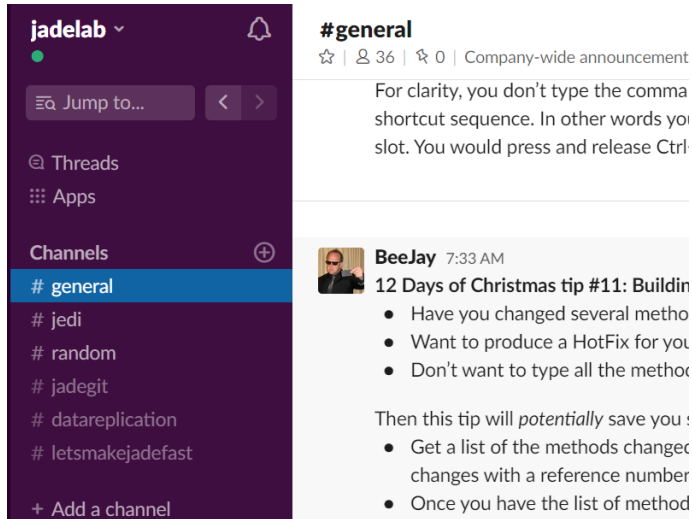
# BeeJay's tips & tricks

- Tip #01 - code folding
- Tip #02 - split editor pane
- Tip #03 - scrolling without losing the carat position
- Tip #04 - hiding the additional selections
- Tip #05 - reuse method windows
- Tip #06 - opening class browser for the current method
- Tip #07 - swapping out of the editor with keyboard
- Tip #08 - using ctrl+6 and ctrl+7 shortcuts
- Tip #09 - getting autocomplete back and moving through camel case words
- Tip #10 - subclassing JadeScript class
- Tip #11 - optimising the find class dialog
- Tip #12 - multiple clipboards
- Tip #13 - abusing parent aspect
- Tip #14 - creating unl files the easy way

# Thank you!



Share your tips and tricks on Slack and Facebook!



A screenshot of a Slack channel named "#general" within a workspace called "jadelab". The channel has 36 members and is marked as a "Company-wide announcement". A message from a user named "BeeJay" at 7:33 AM is shown. The message title is "12 Days of Christmas tip #11: Building" and the content includes a tip about typing comma shortcuts and a list of bullet points.

**#general**  
☆ | 👤 36 | 🔒 0 | Company-wide announcement

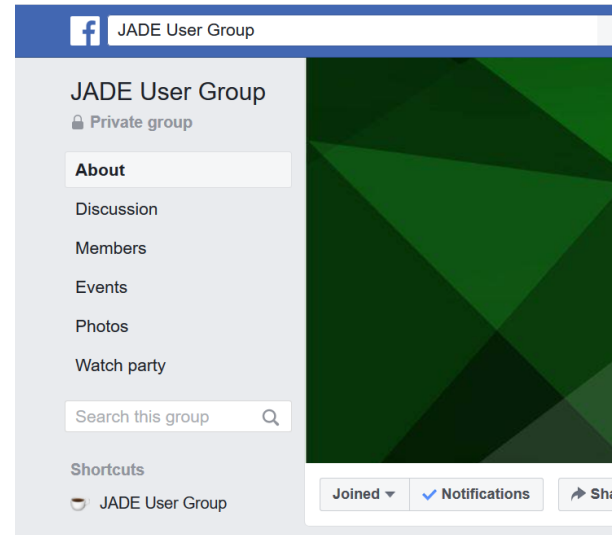
For clarity, you don't type the comma shortcut sequence. In other words you would press and release Ctrl

**BeeJay** 7:33 AM  
**12 Days of Christmas tip #11: Building**

- Have you changed several methods
- Want to produce a HotFix for you
- Don't want to type all the methods

Then this tip will *potentially* save you :

- Get a list of the methods changed with a reference number
- Once you have the list of methods



A screenshot of the Facebook page for the "JADE User Group". The page is for a "Private group" and shows a navigation menu on the left with options like "About", "Discussion", "Members", "Events", "Photos", and "Watch party". There is a search bar and a "Joined" button.

**JADE User Group**  
Private group

**About**

Discussion  
Members  
Events  
Photos  
Watch party

Search this group

Shortcuts  
JADE User Group

Joined ▾ Notifications Share



**That's a wrap!**

Thank you!

**JADE**