



Open Source on z/OS

November 2019



|What I will talk about

- Why open source on the mainframe?
- Rocket's open source offerings
- How is open source different on z/OS?
- Zowe
- Conclusion

Why open source on the mainframe?

| Mainframe remains a critical platform



■ A mature platform

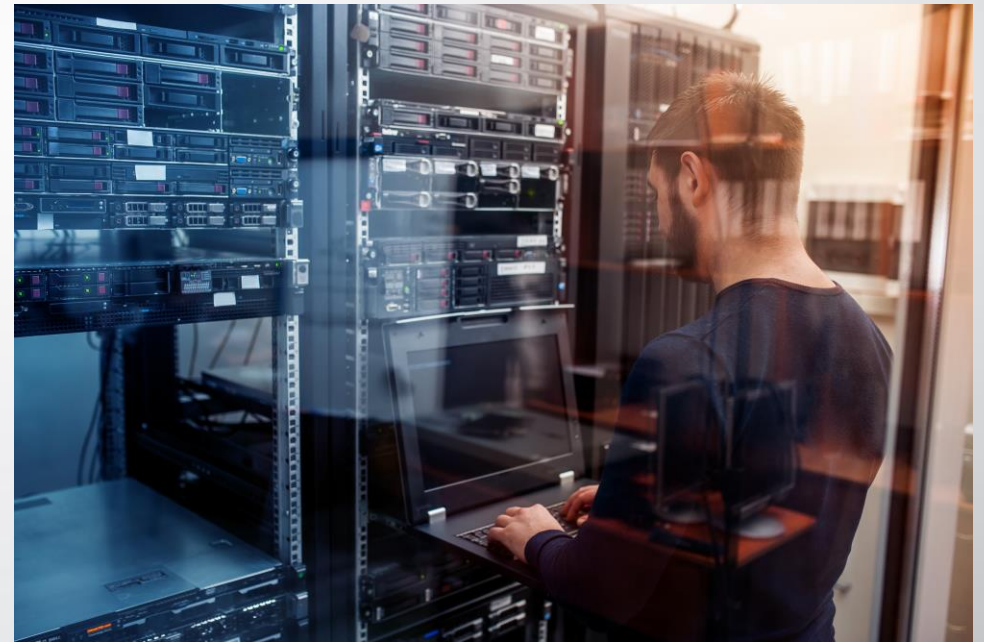
- Impressive uptime
- Excellent transactional performance
- Robust logging

■ It's where the data is

- 92 of the top 100 banks
- 10 of the top 10 airlines
- 23 of the top 25 retailers
- 23 of the top 25 insurance companies

6 The developer problem

- Exclusive group of developers
- High learning curve to join
- Shrinking talent pool

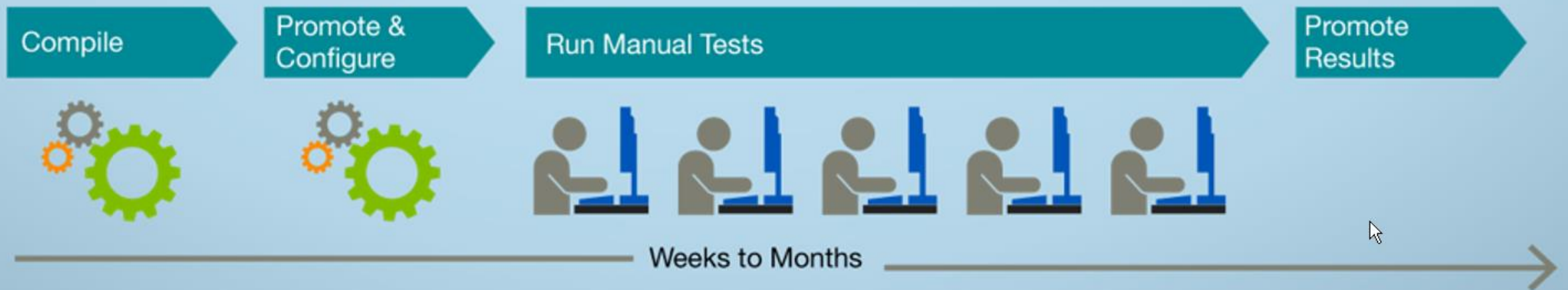


7 The DevOps problem



- Traditional development cycle still the norm

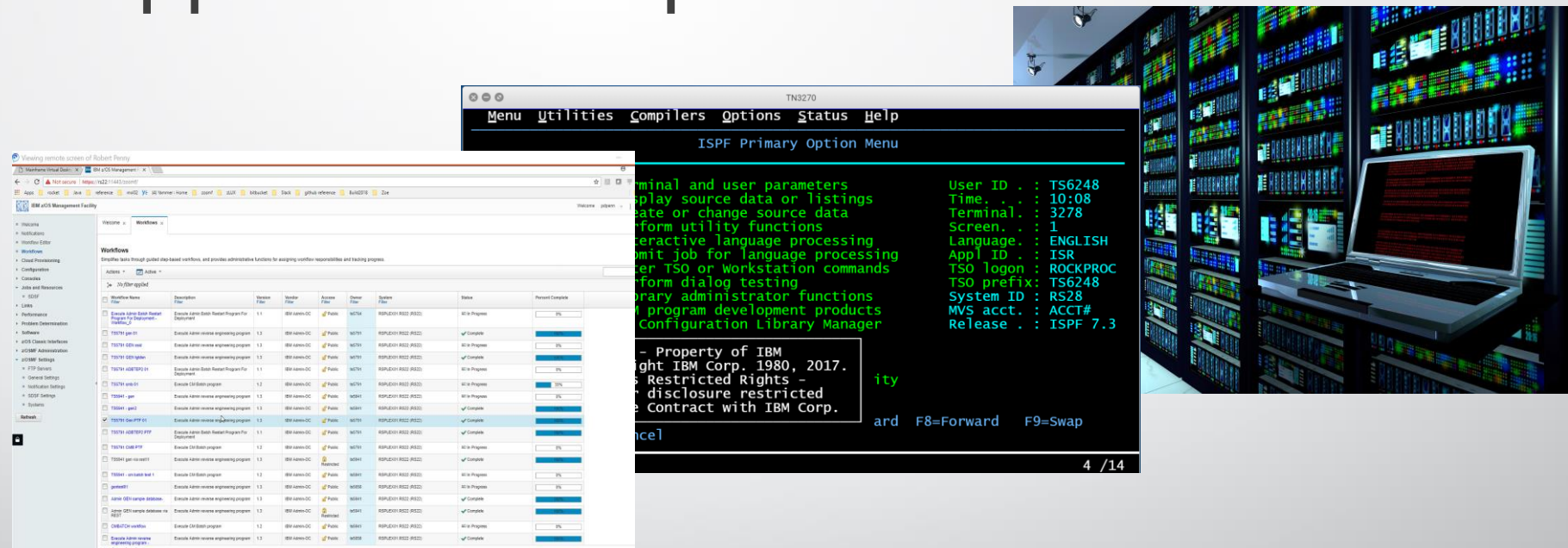
Mainframe Teams Current State



8 The user experience problem



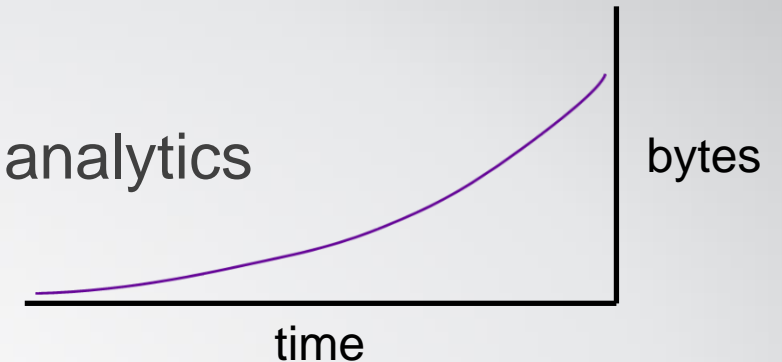
- Green screen technology
- Heterogenous set of browser-based applications
- No cross-application cooperation



The analytics demand problem



- Data growing exponentially and with it demand for analytics



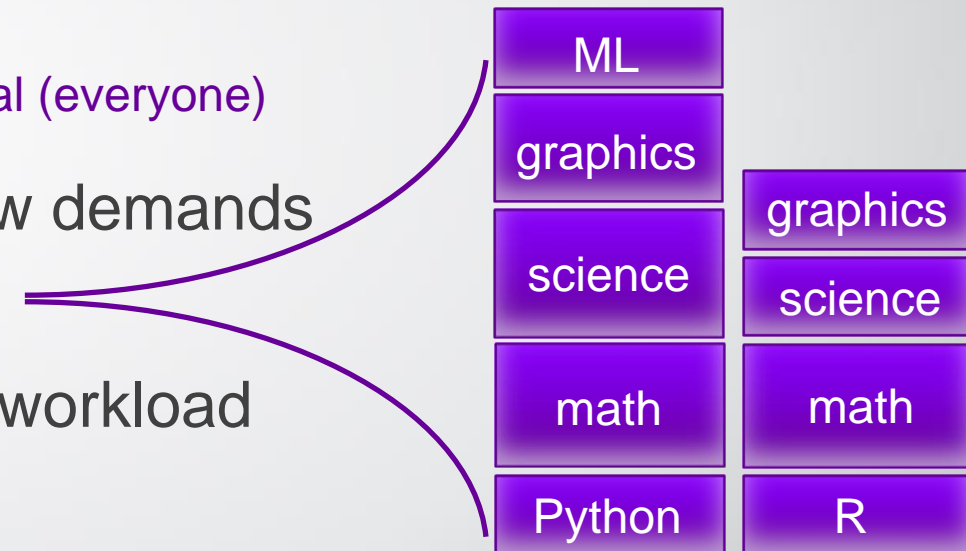
- Demand is for more real-time analytics

Strategic (C-level) → Tactical (management) → Operational (everyone)

- Traditional mainframe languages not ideal for new demands

- Cost is prohibitive for lots of additional analytical workload

Large volume of new DSWL + GPP = \$\$\$



The open source world



- Millions of programmers -- Billions of lines of free code
 - As of May 2019, GitHub reports having over 37 million users^[9] and more than 100 million repositories^[10] (including at least 28 million public repositories),^[11] making it the largest host of source code in the world.^[12] (~45% increase in two years)
- Lower execution costs
- Lower development costs
 - Fast evolving languages, packages, and tools
 - Java, JavaScript, Typescript, Perl, Python, Go, etc
 - SciPy, Scikit-Learn, Spark, Dask
 - git, cURL, PHP, Jupyter, Zeppelin
- Security through transparency

11 | The open source world

■ Modern development cycle

Java / .NET Teams



12 | How to get the benefits? Move the data?



- ETL solutions have serious drawbacks for analytics
 - High cost – high upfront cost and consumes CPU and bandwidth
 - High latency – credit card fraud best done in real-time!
 - Security risks – many data breaches have occurred due to ETL
- Moving data is a non-starter for DevOps
 - The world has moved to git
 - Corporations want a uniform developer experience across all platforms used by an enterprise ...even Z

Rocket's open source offerings on z/OS

Self-Managing
Systems

Security

Self-service
Collaborative BI

★
Machine Learning

★
Automation
Operations
Deployment
DevOps

Analytics
Spark ★
Anaconda

Open Source
Languages and Tools ★

★
API Economy
z/OS Connect

Blockchain

Data
★
Virtualization

Cognitive ★

THE MODERN MAINFRAME

Hybrid Cloud

Discovery



| Open Source Languages on z/OS



- Python 3.7.0* w/ASCII
- R 3.5.2* w/ASCII
- Perl 5.24 w/ASCII
- PHP 7.0.5 w/ASCII



* Most recent versions of Python and R only by special request for the next 3 months while we finish conversion to conda

| Languages are not enough



- Editors (vim)
- Source code control (git)
- Build utilities (autoconf, automake, bison, make, ant*)
- Web development tools (cURL, webalizer, zlib)
- File and Text utilities (diffutils, findutil, m4, sed)
- Compressors and archivers (bzip2, gzip, zip)
- User interfaces (Bash, jupyter)



| Languages and Tools are not enough



- Math and Science:
 - numpy, statsmodel, sympy, scipy
- Analytics and ML:
 - scikit, pandas, blaze, odo, nltk, pyspark, dask
- Presentation:
 - bokeh, seaborn, Django, Flask, cairo
- Package management:
 - conda



Is anyone really using this?



- Yes. In fact, Rocket is building a community...
- <https://forum.rocketsoftware.com/c/rocket-z-os-open-source-languages-tools>
 - 234 topics
 - 723 replies
 - 1M views

The screenshot shows the Rocket Software forum interface. At the top is the Rocket logo and navigation links: Products, Resources, Support, and About Us. Below this is the 'Rocket Software' header with a search icon, a notification bell with 1 red dot, and a user profile icon with 1 blue dot. A blue notification bar asks: 'Do you want live notifications when people reply to your posts? [Enable Notifications.](#)' with a close button. The forum category is 'Open Source Languages and Tools for z/OS', with filters for 'all', 'Latest', 'Unread (20)', and 'Top'. There are buttons for 'Edit', '+ New Topic', and a circular icon. The forum list has columns for 'Topic', 'Replies', 'Views', and 'Activity'. The topics listed are:

Topic	Replies	Views	Activity
★ Downloading Rocket z/OS Open Source Languages and Tools ■ Open Source Languages and Tools for z/OS	60	6.4k	Apr '17
Vim 8.0.22 for z/OS released 4 ■ Open Source Languages and Tools for z/OS vim	34	1.7k	Nov '17
Using git for z/OS with GitHub 20 ■ Open Source Languages and Tools for z/OS git	34	6.5k	Apr 10
PHP 7 character set issues 2 ■ PHP	31	2.3k	Nov '18
Output from GIT always coming back in ASCII ■ Open Source Languages and Tools for z/OS	25	1.4k	Jul '18

| Solution for analytics demand problem

- move the compute to the data

- Port open source to z/OS
 - Python and R -- #1 and #2 in data science (after Excel)
 - Spark
 - Anaconda
 - ...and Bash
- IBM MDS for Apache Spark enables easy (SQL) access to data
 - CICS
 - IMS
 - SMF
 - DB2
 - VSAM

How is open source
different on z/OS?
(example: git)

| Mostly it is about character encoding?



■ z/OS Enhanced ASCII

- Source code is ASCII encoded
- All open source is compiled with `-qascii` option
- All open source is ported to support CCSID tagged files
- Automatic conversion is enabled via `_BPXK_AUTOCVT`
- Untagged files are presumed to be EBCDIC

| Encoding Solution for z/OS git



```
# This sets the default encoding for files that do NOT match any of the
# following patterns. Many z/OS users will want the bulk of their files
# as EBCDIC.

* git-encoding=iso8859-1 working-tree-encoding=ibm-1047

# Some of the git configuration files MUST be ASCII.

.gitattributes      git-encoding=iso8859-1 zos-working-tree-encoding=iso8859-1
.gitignore          git-encoding=iso8859-1 zos-working-tree-encoding=iso8859-1

# Some files are not text; leave them untouched, and tagged as binary files.

*.jpg      git-encoding=BINARY      zos-working-tree-encoding=BINARY
*.png      git-encoding=BINARY      zos-working-tree-encoding=BINARY
*.gif      git-encoding=BINARY      zos-working-tree-encoding=BINARY
*.zip      git-encoding=BINARY      zos-working-tree-encoding=BINARY
```

z DevOps Pipeline at Rocket using Open Source

zDevOps at Rocket



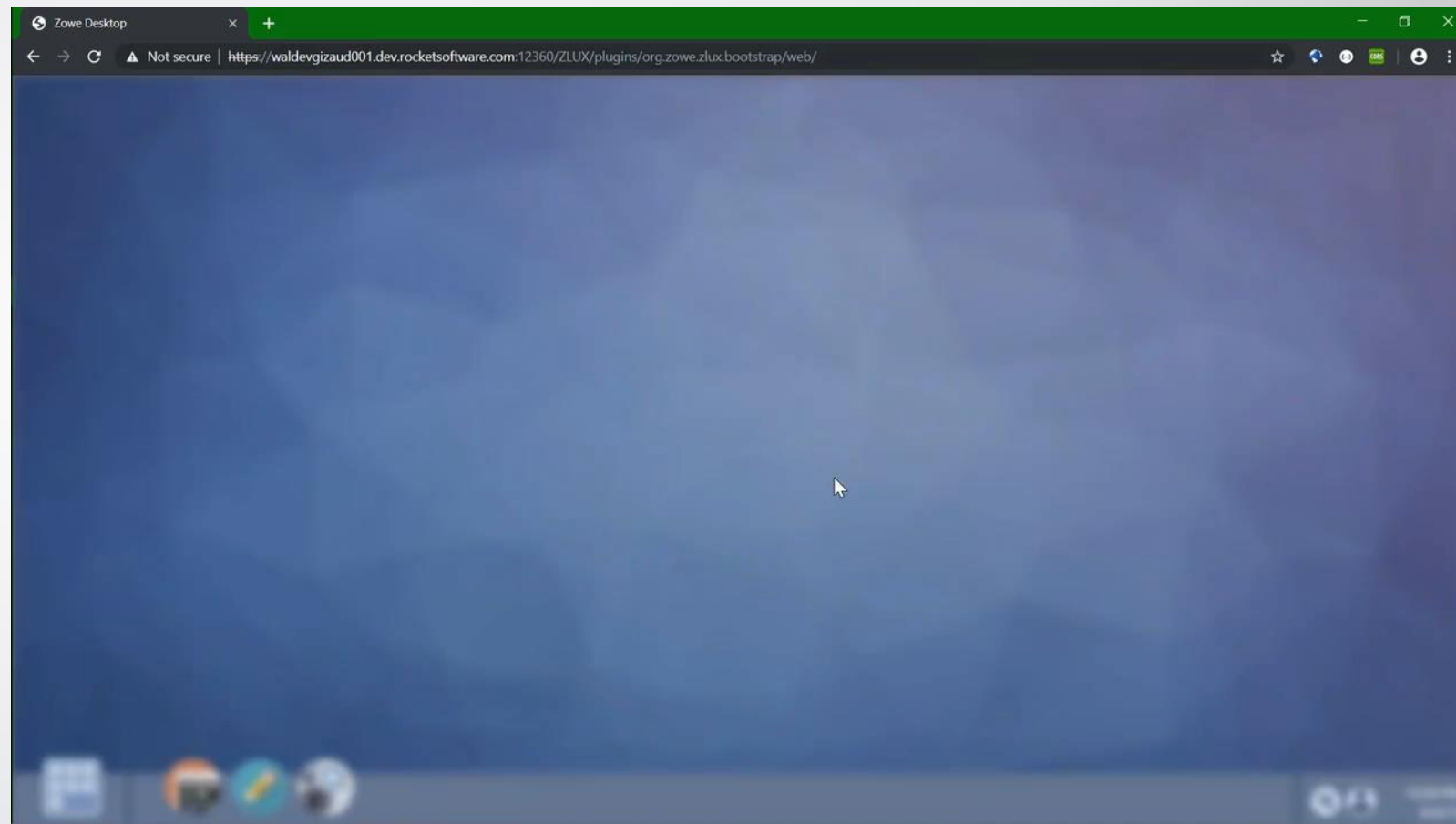
- Conductor: Jenkins on Linux (off mainframe)
- SCM: Git Server (BitBucket off mainframe) & **z/OS Git Client**
 - Source cloned from BitBucket to z/OS Unix System Services (USS)
- Build Scripting:
 - Moving files between USS and MVS (Partitioned Data Sets)
 - Executing builds
 - **z/OS ant**
 - **z/OS bash**
 - **z/OS python**
- Automated testing and scanning:
 - **z/OS Python**
- (planned) Artifact Repository: Artifactory (off mainframe)

Introducing Zowe

The Opportunity



- Create a unified environment
 - Homogenous desktop-like platform
 - Modern look-and-feel
 - Single sign-on
 - Cross-application launch & communication between vendors
 - 0 install, cloud-like UI
 - Access to legacy web applications, terminals
 - Tools to port & modernize in favorite web technology



A Brand New Experience



The screenshot displays a comprehensive software interface for managing and monitoring systems. The interface is divided into several panes:

- TN3270 terminal:** Shows a command prompt with various system commands and their outputs, including storage group space summary and lowest volume free space.
- TEP (Tape Element Processor):** Displays a bar chart titled "Storage Groups with > 80% used space" showing the usage of various storage groups. Below the chart is a table with columns: Group Name, Total Volumes, Total Space MB, Free Space MB, Free Space Percent, Largest Free Extent MB, High Volume Fragmentation Index, and Used Space %.
- JES Explorer:** A window for managing JES (Job Entry Subsystem) jobs, showing a list of jobs with columns for Job Name, Job ID, Owner, and Status. It includes a "Purge Job" button.
- Web Browser:** Displays the "Zowe Desktop" page, which includes a "Not Secure" warning and a URL. The page features a "Discovery Service" section with "API Gateway" and "API Catalog" links.
- TN3270 terminal (bottom right):** Shows a command prompt with various system commands and their outputs, including storage group space summary and lowest volume free space.

The Opportunity



- Attract the new generation of developers
 - Best-of-breed web technologies
 - Simplified application architecture
 - Self documenting REST-APIs
 - Reduced learning curve
 - Open source effort



The Opportunity



- Modernize and enhance existing applications
- Free applications with paying upgrades
- Converge onto a single platform
 - Unified technological stack



The Opportunity



- Increase application development speed
 - Fast development through standard web-app architecture
- Improve release agility
 - Faster delivery of small targeted changes within an app
 - In-between-release delivery



- Improve responsiveness to customer needs
 - Make, test, and deploy changes in days
 - Create, test, and deploy custom apps in weeks
- Involve customers in the development effort
 - Customers create their own applications and contribute them back to the community
 - Direct insight into their needs and perspectives

The Plan

- Create a web-based desktop environment
 - Rocket's contribution to Zowe
- Expose core z/OS functionalities through REST APIs
 - The empowering move that makes it easy for web developers to master the mainframe
- Plug into the z/OS security model
 - A no-compromise zone for mainframe operators

How We Get There



■ Drive adoption

- Commercial offerings on top of Zowe
- Customer Advisory Board under OMP

■ Provide support

- Training material and documentation
- Commercial support

Open Source Transformation



■ Rocket

- Founding member of a new Open Mainframe Project
 - <https://www.openmainframeproject.org/>
- Member of the OMP Governing Board
- Part of the Linux Foundation



■ Partnership with IBM and Broadcom

The dawn of a new mainframe ecosystem

Zowe

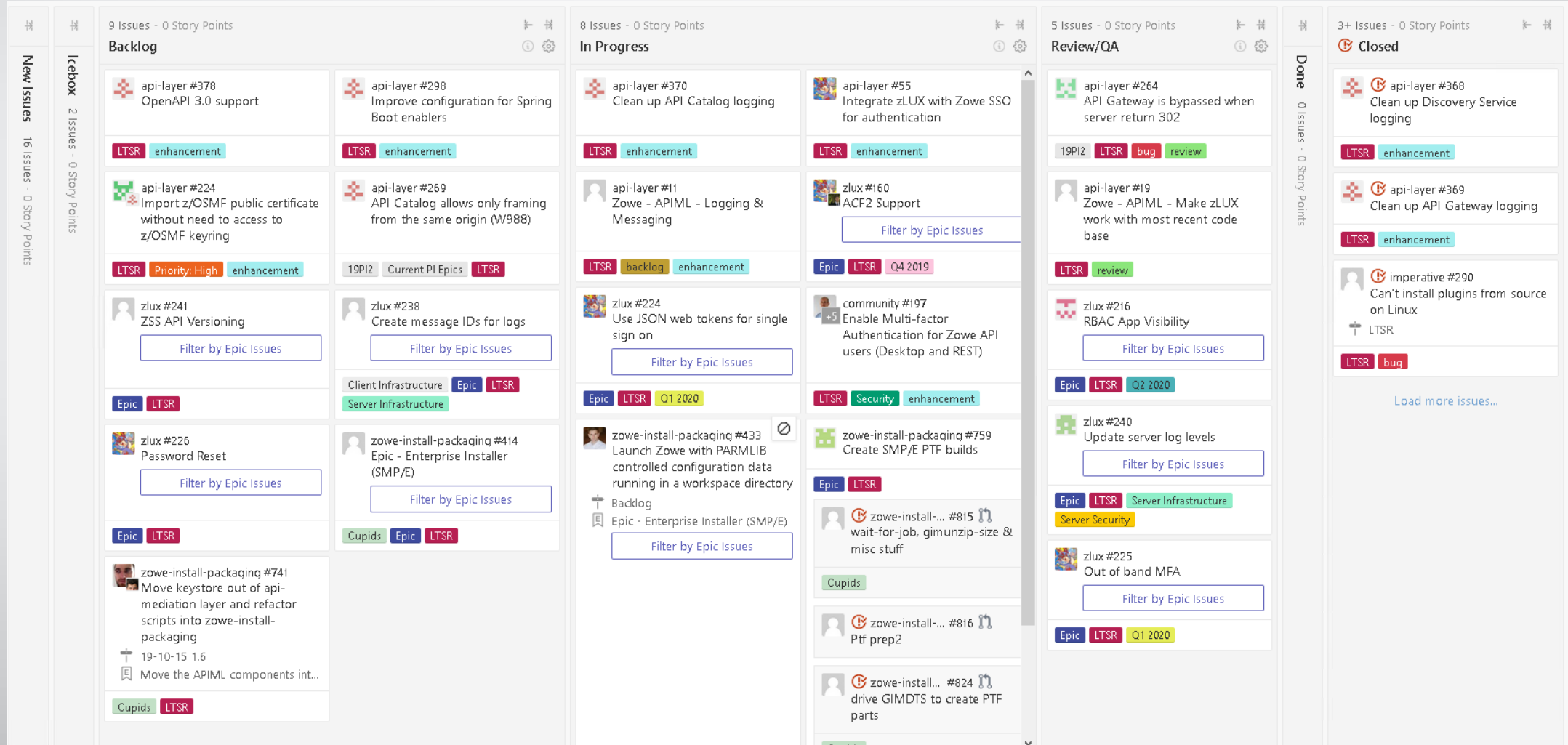
What to look for in 2020

Solution-based z/OS open source bundles

- Bundles of ports that work together to solve common problems (e.g. DevOps)
- Easier user experience via conda and packaging and delivery
- 24x7 support offerings to help you succeed in production
- Bundles include examples and extensions specific to DevOps
- Services offerings to get you started

Zowe is now mature: v1.x LTS 2020, v2 starts 2020

See <https://zowe.org> & <https://github.com/zowe> & [Zenhub](#)



The screenshot displays the Zenhub interface with four issue boards. Each board has a title, a count of issues, and a list of individual issues with their details.

Backlog

9 Issues - 0 Story Points

- api-layer #378: OpenAPI 3.0 support (LTSR, enhancement)
- api-layer #298: Improve configuration for Spring Boot enablers (LTSR, enhancement)
- api-layer #224: Import z/OSMF public certificate without need to access to z/OSMF keyring (LTSR, Priority: High, enhancement)
- api-layer #269: API Catalog allows only framing from the same origin (W988) (19PI2, Current PI Epics, LTSR)
- zlux #241: ZSS API Versioning (Filter by Epic Issues, Epic, LTSR)
- zlux #238: Create message IDs for logs (Filter by Epic Issues, Client Infrastructure, Epic, LTSR, Server Infrastructure)
- zlux #226: Password Reset (Filter by Epic Issues, Epic, LTSR)
- zowe-install-packaging #414: Epic - Enterprise Installer (SMP/E) (Filter by Epic Issues, Cupids, Epic, LTSR)
- zowe-install-packaging #741: Move keystore out of api-mediation layer and refactor scripts into zowe-install-packaging (19-10-15 1.6, Move the APIML components int..., Cupids, LTSR)

In Progress

8 Issues - 0 Story Points

- api-layer #370: Clean up API Catalog logging (LTSR, enhancement)
- api-layer #11: Zowe - APIML - Logging & Messaging (LTSR, backlog, enhancement)
- zlux #224: Use JSON web tokens for single sign on (Filter by Epic Issues, Epic, LTSR, Q1 2020)
- zowe-install-packaging #433: Launch Zowe with PARMLIB controlled configuration data running in a workspace directory (Backlog, Epic - Enterprise Installer (SMP/E), Filter by Epic Issues)
- api-layer #55: Integrate zLUX with Zowe SSO for authentication (LTSR, enhancement)
- zlux #160: ACF2 Support (Filter by Epic Issues, Epic, LTSR, Q4 2019)
- community #197: Enable Multi-factor Authentication for Zowe API users (Desktop and REST) (LTSR, Security, enhancement)
- zowe-install-packaging #759: Create SMP/E PTF builds (Epic, LTSR)
- zowe-install-... #815: wait-for-job, gimunzip-size & misc stuff (Cupids)
- zowe-install-... #816: Ptf prep2 (Cupids)
- zowe-install-... #824: drive GIMDTS to create PTF parts (Cupids)

Review/QA

5 Issues - 0 Story Points

- api-layer #264: API Gateway is bypassed when server return 302 (19PI2, LTSR, bug, review)
- api-layer #19: Zowe - APIML - Make zLUX work with most recent code base (LTSR, review)
- zlux #216: RBAC App Visibility (Filter by Epic Issues, Epic, LTSR, Q2 2020)
- zlux #240: Update server log levels (Filter by Epic Issues, Epic, LTSR, Server Infrastructure, Server Security)
- zlux #225: Out of band MFA (Filter by Epic Issues, Epic, LTSR, Q1 2020)

Done

3+ Issues - 0 Story Points

- api-layer #368: Clean up Discovery Service logging (LTSR, enhancement)
- api-layer #369: Clean up API Gateway logging (LTSR, enhancement)
- imperative #290: Can't install plugins from source on Linux (LTSR, bug)

Conclusion

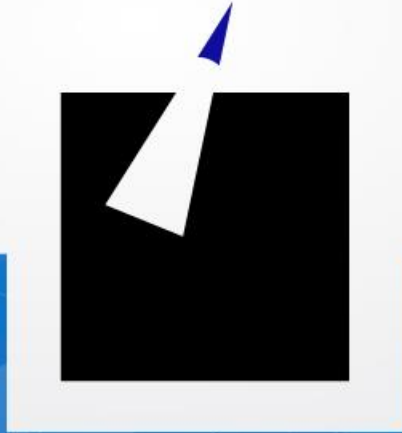


Conclusion

Re-enabling new line of business
application development on z/OS



IT'S NOT ROCKET SCIENCE.
IT'S ROCKET SOFTWARE.

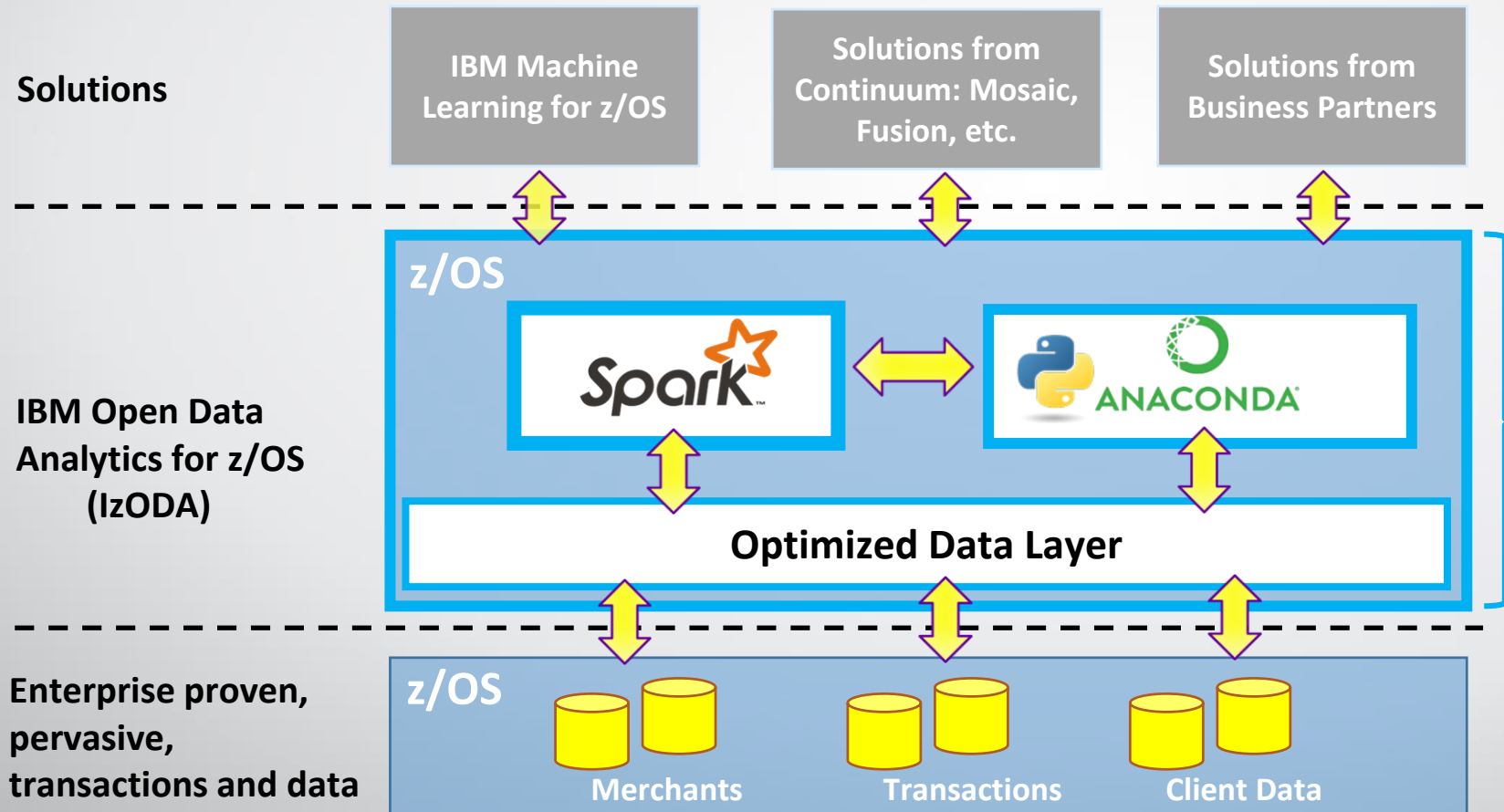


Backup Slides

IBM Open Data Analytics for z/OS V1.1

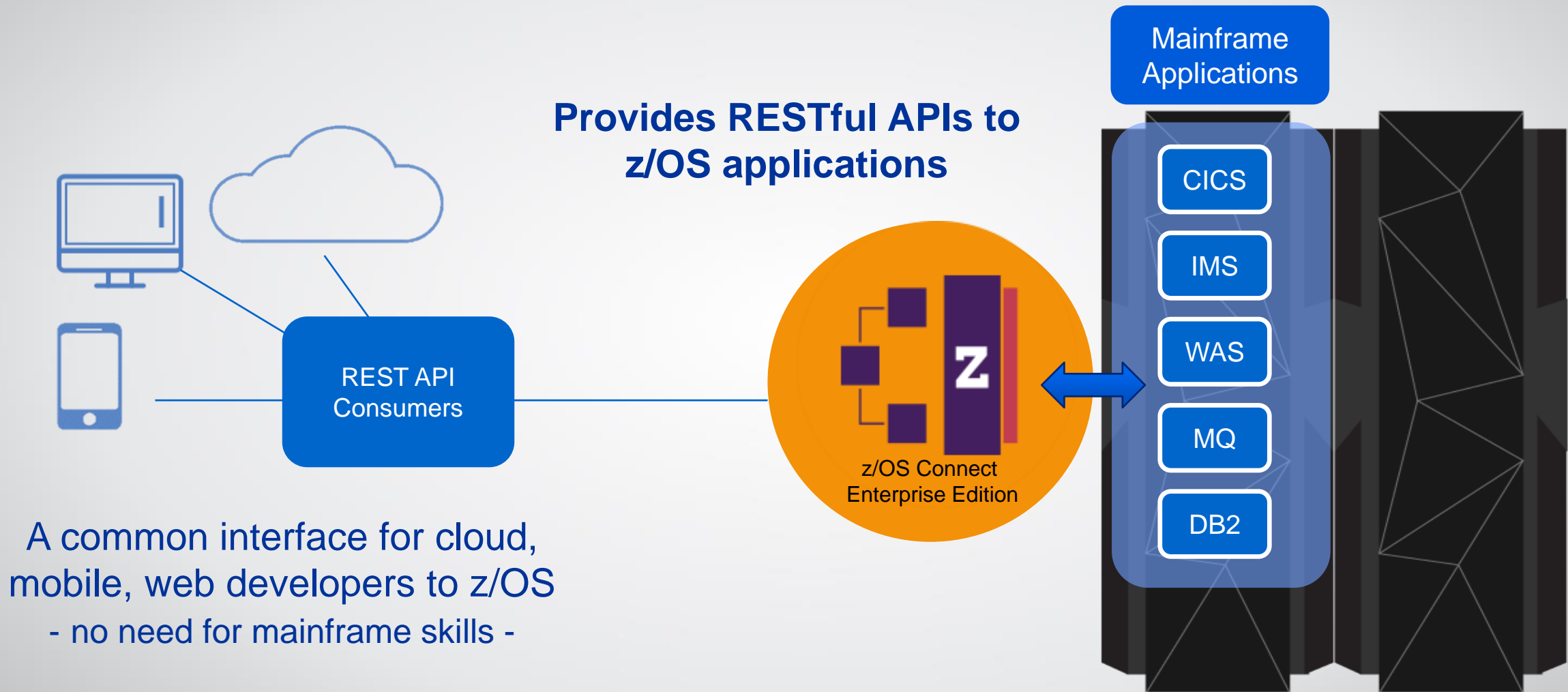


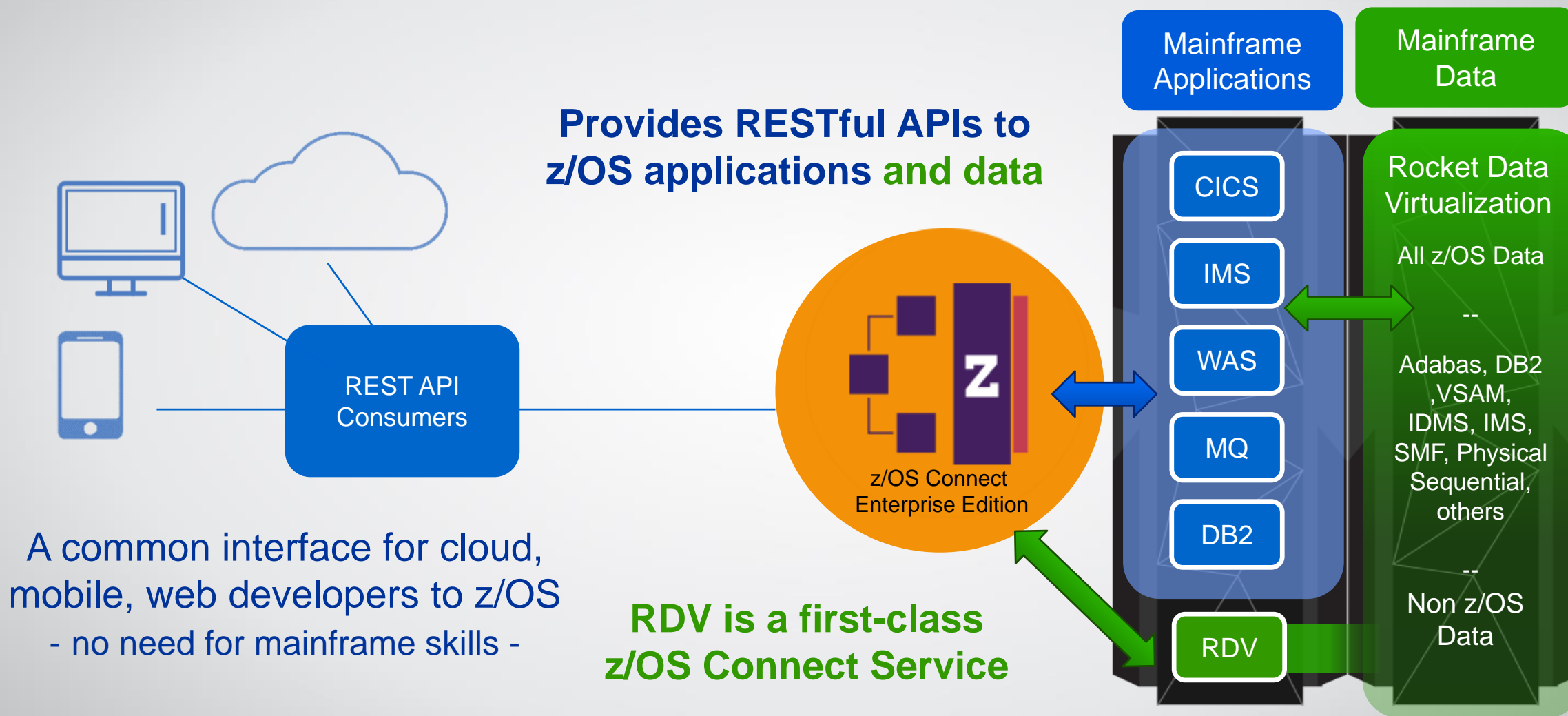
$$\text{IBM} + \text{Rocket} + \text{CONTINUUM ANALYTICS} = \text{https://izoda.github.io/}$$



- Establish rich runtimes for analytics co-located with IBM Z data
- Leverage leading open source technologies
- Enable advanced solutions from IBM and partners
- Integrate and differentiate with z/OS security & workload

RDV for Modernization





Offline “Demo”

Demonstrate:

- Setting up Anaconda
- Installing git
- Cloning a github project
- A jupyter notebook
 - written in python
 - getting data via the optimized data layer
 - using data science packages to process, analyze, and visualize in order to create a prediction model for that data.

| Logon via Putty

A screenshot of a PuTTY terminal window. The title bar reads "rs22.rocketsoftware.com - PuTTY". The terminal text shows the login process: "Using username 'PDFAND'." followed by "Authenticating with public key 'peter-window-public-key' from agent". The prompt "RS22 /u/pdfand>" is shown with a green cursor. The terminal has a light blue background and standard window controls.

```
rs22.rocketsoftware.com - PuTTY
Using username "PDFAND".
Authenticating with public key "peter-window-public-key" from agent
RS22 /u/pdfand> 
```

| Start Bash



```
rs22.rocketsoftware.com - PuTTY
Using username "PDFAND".
Authenticating with public key "peter-window-public-key" from agent
RS22 /u/pdfand> cd /u/pdharr/IBM/izoda/anaconda
RS22 /u/pdharr/IBM/izoda/anaconda> bin/bash
PDFANDA@RS22:/u/pdharr/IBM/izoda/anaconda>
```

A screenshot of a PuTTY terminal window titled "rs22.rocketsoftware.com - PuTTY". The window shows an SSH session where the user "PDFAND" is authenticated using a public key. The user then runs the command "cd /u/pdharr/IBM/izoda/anaconda" and "bin/bash" to start a Bash shell. The prompt changes from "RS22" to "PDFANDA@RS22" and the directory is "/u/pdharr/IBM/izoda/anaconda". A green cursor is visible at the end of the prompt.

| Set up Anaconda



```
rs22.rocketsoftware.com - PuTTY
Using username "PDFAND".
Authenticating with public key "peter-window-public-key" from agent
RS22 /u/pdfand> cd /u/pdharr/IBM/izoda/anaconda
RS22 /u/pdharr/IBM/izoda/anaconda> bin/bash
PDFANDA@RS22:/u/pdharr/IBM/izoda/anaconda> source bin/activate
(root) PDFANDA@RS22:/u/pdharr/IBM/izoda/anaconda> █
```

Create your sandbox directory



```
rs22.rocketsoftware.com - PuTTY
Using username "PDFAND".
Authenticating with public key "peter-window-public-key" from agent
RS22 /u/pdfand> cd /u/pdharr/IBM/izoda/anaconda
RS22 /u/pdharr/IBM/izoda/anaconda> bin/bash
PDFANDA@RS22:/u/pdharr/IBM/izoda/anaconda> source bin/activate
(root) PDFANDA@RS22:/u/pdharr/IBM/izoda/anaconda> cd
(root) PDFANDA@RS22:~> mkdir anthem
(root) PDFANDA@RS22:~> cd anthem
(root) PDFANDA@RS22:~/anthem> █
```

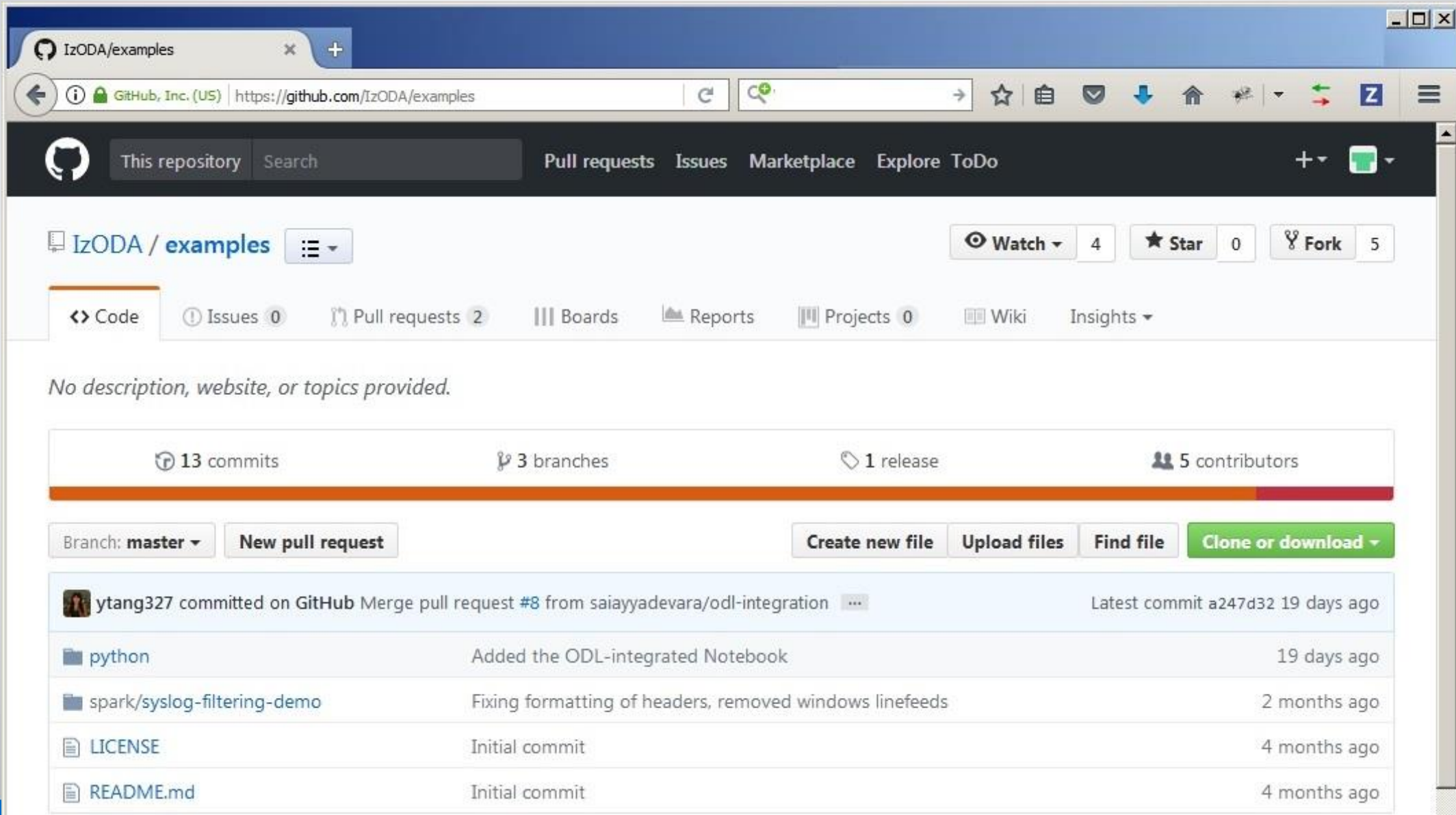

Install latest Bash, git, and Perl



```
rs22.rocketsoftware.com - PuTTY
Using username "PDFAND".
Authenticating with public key "peter-window-public-key" from agent
RS22 /u/pdfand> cd /u/pdharr/IBM/izoda/anaconda
RS22 /u/pdharr/IBM/izoda/anaconda> bin/bash
PDFANDA@RS22:/u/pdharr/IBM/izoda/anaconda> source bin/activate
(root) PDFANDA@RS22:/u/pdharr/IBM/izoda/anaconda> cd
(root) PDFANDA@RS22:~> mkdir anthem
(root) PDFANDA@RS22:~> cd anthem
(root) PDFANDA@RS22:~/anthem> conda install bash=4.3 git perl
Fetching package metadata .....
Solving package specifications: .

# All requested packages already installed.
# packages in environment at /u/pdharr/IBM/izoda/anaconda:
#
bash                4.3.46                18      IzODA
git                  2.3.5                  13      IzODA
perl                  5.24.0                  5      IzODA
(root) PDFANDA@RS22:~/anthem> █
```

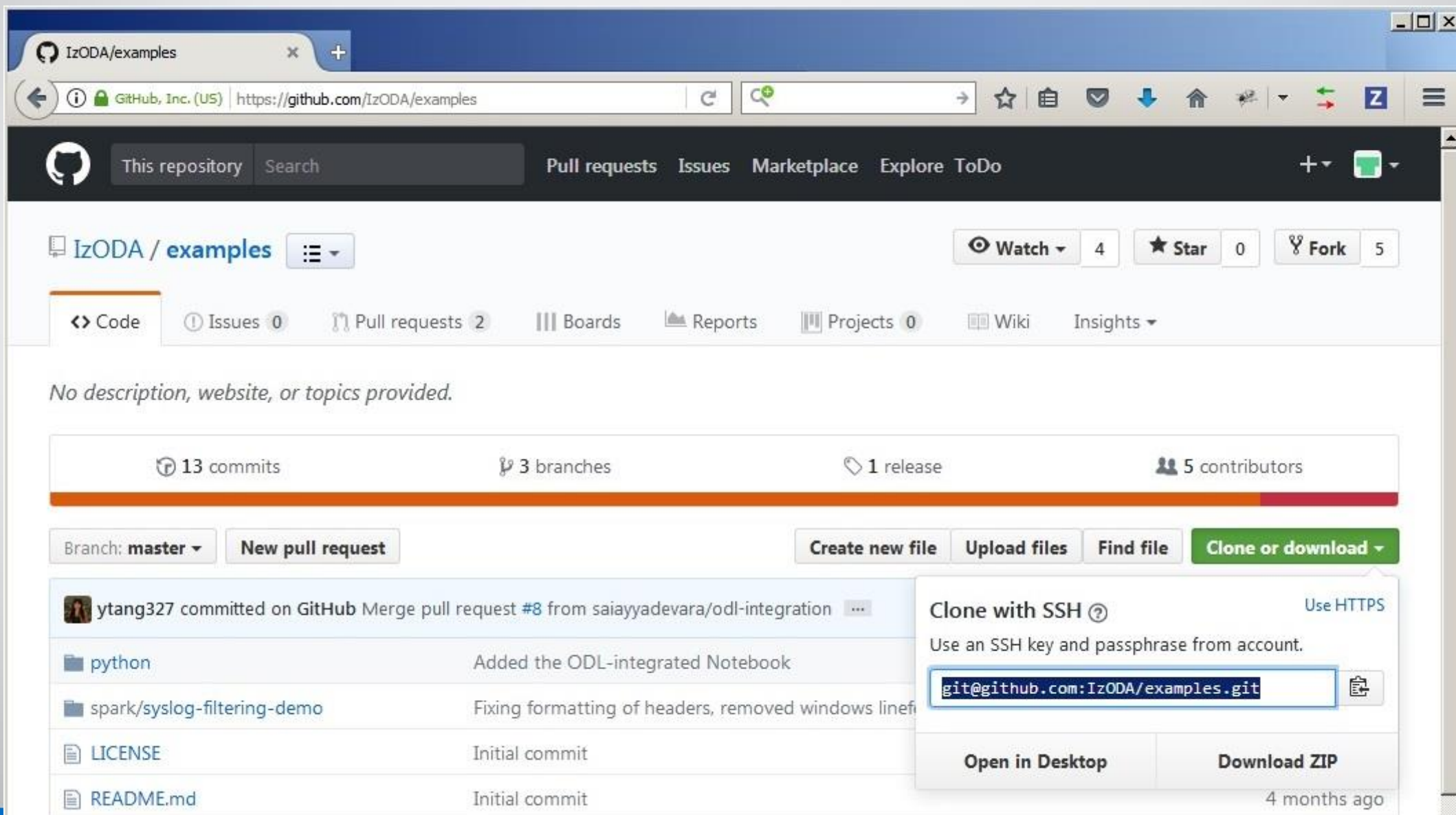
Locate code on github



The screenshot shows the GitHub repository page for `IzODA/examples`. The browser address bar shows the URL `https://github.com/IzODA/examples`. The repository page includes a header with the repository name, a search bar, and navigation links for Pull requests, Issues, Marketplace, Explore, and ToDo. Below the header, the repository name `IzODA / examples` is displayed, along with buttons for Watch (4), Star (0), and Fork (5). A secondary navigation bar shows tabs for Code, Issues (0), Pull requests (2), Boards, Reports, Projects (0), Wiki, and Insights. A message states "No description, website, or topics provided." Below this, a summary bar shows 13 commits, 3 branches, 1 release, and 5 contributors. A row of buttons includes "Branch: master", "New pull request", "Create new file", "Upload files", "Find file", and "Clone or download". The commit history table lists the following entries:

Commit	Message	Time
ytang327 committed on GitHub Merge pull request #8 from saiayyadevara/odl-integration		Latest commit a247d32 19 days ago
python	Added the ODL-integrated Notebook	19 days ago
spark/syslog-filtering-demo	Fixing formatting of headers, removed windows linefeeds	2 months ago
LICENSE	Initial commit	4 months ago
README.md	Initial commit	4 months ago

Copy a clone reference to clipboard



The screenshot shows the GitHub interface for the repository 'IzODA/examples'. The browser address bar displays the URL 'https://github.com/IzODA/examples'. The repository page includes a navigation bar with links for 'This repository', 'Search', 'Pull requests', 'Issues', 'Marketplace', 'Explore', and 'ToDo'. Below the navigation bar, the repository name 'IzODA / examples' is shown, along with statistics: 4 watches, 0 stars, and 5 forks. A tabbed interface allows switching between 'Code', 'Issues', 'Pull requests', 'Boards', 'Reports', 'Projects', 'Wiki', and 'Insights'. The 'Code' tab is active, displaying a commit history table with columns for the commit author, the commit message, and the commit type. The commit history table shows four commits: a merge pull request, and three initial commits for 'python', 'spark/syslog-filtering-demo', and 'LICENSE' and 'README.md'. A 'Clone or download' button is visible, which has opened a dropdown menu. The dropdown menu offers two options: 'Clone with SSH' and 'Use HTTPS'. The 'Clone with SSH' option is selected, and the SSH clone reference 'git@github.com:IzODA/examples.git' is displayed in a text box, which is highlighted with a blue border. A clipboard icon is next to the text box, indicating that the reference has been copied to the clipboard. Below the text box, there are buttons for 'Open in Desktop' and 'Download ZIP'. The 'Download ZIP' button is disabled, and the text '4 months ago' is visible below it.

13 commits 3 branches 1 release 5 contributors

Branch: master New pull request Create new file Upload files Find file Clone or download

Commit	Message	Type
ytang327 committed on GitHub Merge pull request #8 from saiyyadevara/odl-integration		Merge
python	Added the ODL-integrated Notebook	Initial commit
spark/syslog-filtering-demo	Fixing formatting of headers, removed windows linef	Initial commit
LICENSE	Initial commit	Initial commit
README.md	Initial commit	Initial commit

Clone with SSH Use HTTPS

Use an SSH key and passphrase from account.

git@github.com:IzODA/examples.git

Open in Desktop Download ZIP

4 months ago

Clone repository



```
rs22.rocketsoftware.com - PuTTY
Using username "PDFAND".
Authenticating with public key "peter-window-public-key" from agent
RS22 /u/pdfand> cd /u/pdharr/IBM/izoda/anaconda
RS22 /u/pdharr/IBM/izoda/anaconda> bin/bash
PDFANDA@RS22:/u/pdharr/IBM/izoda/anaconda> source bin/activate
(root) PDFANDA@RS22:/u/pdharr/IBM/izoda/anaconda> cd
(root) PDFANDA@RS22:~> mkdir anthem
(root) PDFANDA@RS22:~> cd anthem
(root) PDFANDA@RS22:~/anthem> conda install bash=4.3 git perl
Fetching package metadata .....
Solving package specifications: .

# All requested packages already installed.
# packages in environment at /u/pdharr/IBM/izoda/anaconda:
#
bash                4.3.46                18      IzODA
git                  2.3.5                  13      IzODA
perl                  5.24.0                  5      IzODA
(root) PDFANDA@RS22:~/anthem> git clone git@github.com:IzODA/examples.git
```

| Ready to begin work!



```
rs22.rocketsoftware.com - PuTTY
RS22 /u/pdfand> cd /u/pdharr/IBM/izoda/anaconda
RS22 /u/pdharr/IBM/izoda/anaconda> bin/bash
PDFANDA@RS22:/u/pdharr/IBM/izoda/anaconda> source bin/activate
(root) PDFANDA@RS22:/u/pdharr/IBM/izoda/anaconda> cd
(root) PDFANDA@RS22:~> mkdir anthem
(root) PDFANDA@RS22:~> cd anthem
(root) PDFANDA@RS22:~/anthem> conda install bash=4.3 git perl
Fetching package metadata .....
Solving package specifications: .

# All requested packages already installed.
# packages in environment at /u/pdharr/IBM/izoda/anaconda:
#
bash                4.3.46                18      IzODA
git                  2.3.5                  13      IzODA
perl                  5.24.0                  5      IzODA
(root) PDFANDA@RS22:~/anthem> git clone git@github.com:IzODA/examples.git
Cloning into 'examples'...
remote: Counting objects: 54, done.
remote: Total 54 (delta 0), reused 0 (delta 0), pack-reused 54
Receiving objects: 100% (54/54), 336.59 KiB | 0 bytes/s, done.
Resolving deltas: 100% (9/9), done.
Checking connectivity... done.
(root) PDFANDA@RS22:~/anthem> 
```


What can one do with this?



Home RPAC-May-2017

rs99.rocketsoftware.com:8888/notebooks/RPAC-May-2017.ipynb

jupyter RPAC-May-2017 Last Checkpoint: 05/04/2017 (autosaved) Logout

File Edit View Insert Cell Kernel Help Python 2

Transforming Big Data Assets into Actionable Intelligence

Connect to virtual DB via Rocket Data Virtualization

In [22]:

```
# Set ODL identifiers
ODL_HOST = "rs99.rocketsoftware.com" # "rs28.rocketsoftware.com"
ODL_PORT = 1307 # 1207
ODL_DB = "DVS"

# NOT a Mongo source
# Leverage Mongo Client API as alpha workaround to connect python to ODL
from pymongo import MongoClient
client = MongoClient(ODL_HOST, ODL_PORT)

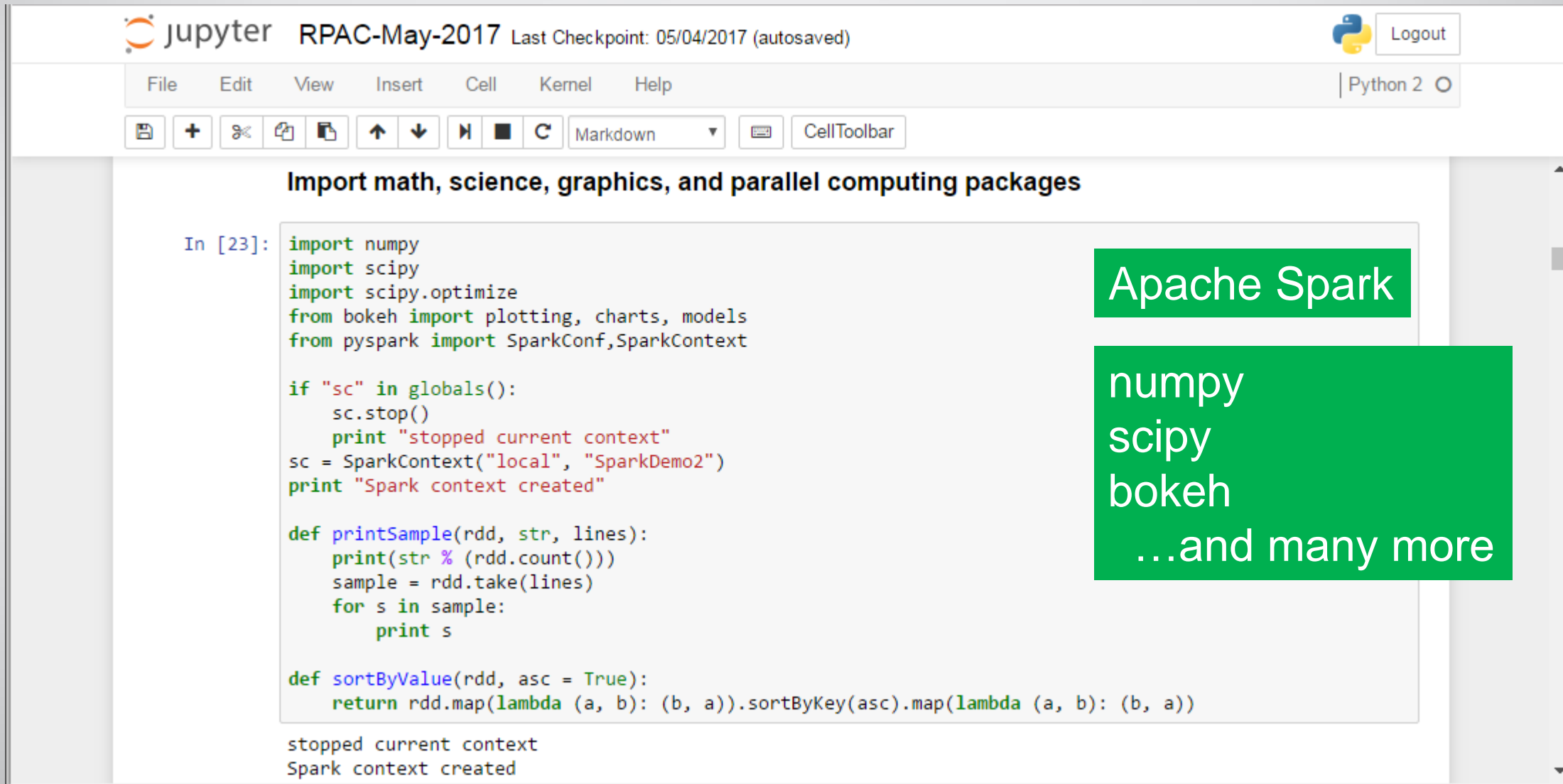
# Get database
db = client[ODL_DB]

print "Connected to %s at %s:%i" % (ODL_DB, ODL_HOST, ODL_PORT)
```

Connected to DVS at rs99.rocketsoftware.com:1307

Python + RDV

What can one do with this?



The image shows a Jupyter Notebook interface with the title "RPAC-May-2017" and a "Last Checkpoint: 05/04/2017 (autosaved)" message. The interface includes a menu bar (File, Edit, View, Insert, Cell, Kernel, Help) and a toolbar with icons for saving, adding cells, and running code. The code cell is titled "Import math, science, graphics, and parallel computing packages" and contains the following Python code:

```
In [23]: import numpy
import scipy
import scipy.optimize
from bokeh import plotting, charts, models
from pyspark import SparkConf, SparkContext

if "sc" in globals():
    sc.stop()
    print "stopped current context"
sc = SparkContext("local", "SparkDemo2")
print "Spark context created"

def printSample(rdd, str, lines):
    print(str % (rdd.count()))
    sample = rdd.take(lines)
    for s in sample:
        print s

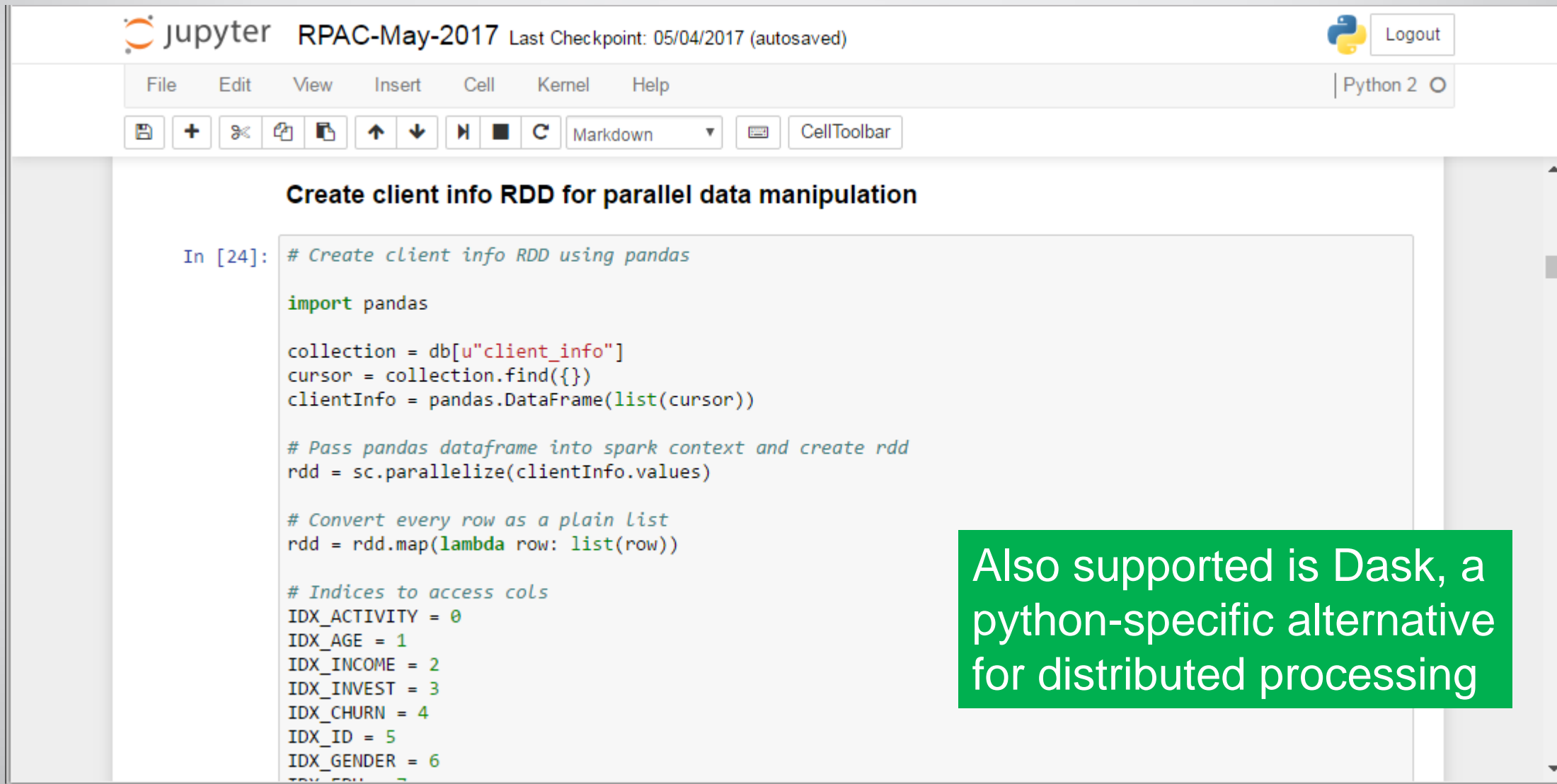
def sortByValue(rdd, asc = True):
    return rdd.map(lambda (a, b): (b, a)).sortByKey(asc).map(lambda (a, b): (b, a))
```

The output of the code is:

```
stopped current context
Spark context created
```

On the right side of the code cell, there are two green boxes. The top box contains the text "Apache Spark". The bottom box contains the text "numpy", "scipy", "bokeh", and "...and many more".

What can one do with this?



The image shows a Jupyter Notebook interface. At the top, the header includes the Jupyter logo, the text "jupyter RPAC-May-2017", and "Last Checkpoint: 05/04/2017 (autosaved)". On the right, there is a "Logout" button and a "Python 2" selector. Below the header is a menu bar with "File", "Edit", "View", "Insert", "Cell", "Kernel", and "Help". A toolbar contains icons for saving, adding, deleting, and other actions, along with a "CellToolbar" button. The main area of the notebook has a title "Create client info RDD for parallel data manipulation" and a code cell. The code cell is labeled "In [24]:" and contains the following Python code:

```
# Create client info RDD using pandas

import pandas

collection = db[u"client_info"]
cursor = collection.find({})
clientInfo = pandas.DataFrame(list(cursor))

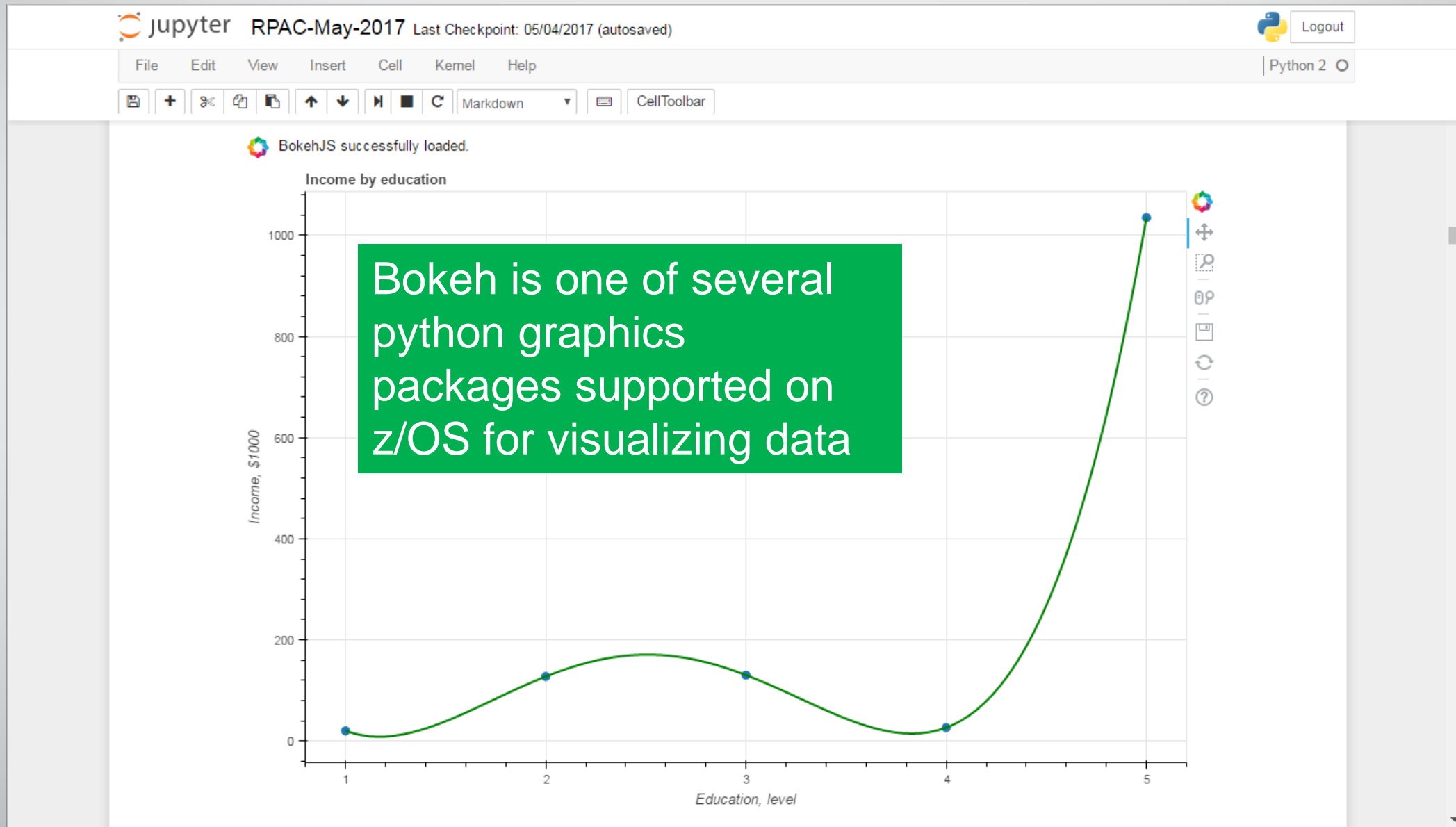
# Pass pandas dataframe into spark context and create rdd
rdd = sc.parallelize(clientInfo.values)

# Convert every row as a plain list
rdd = rdd.map(lambda row: list(row))

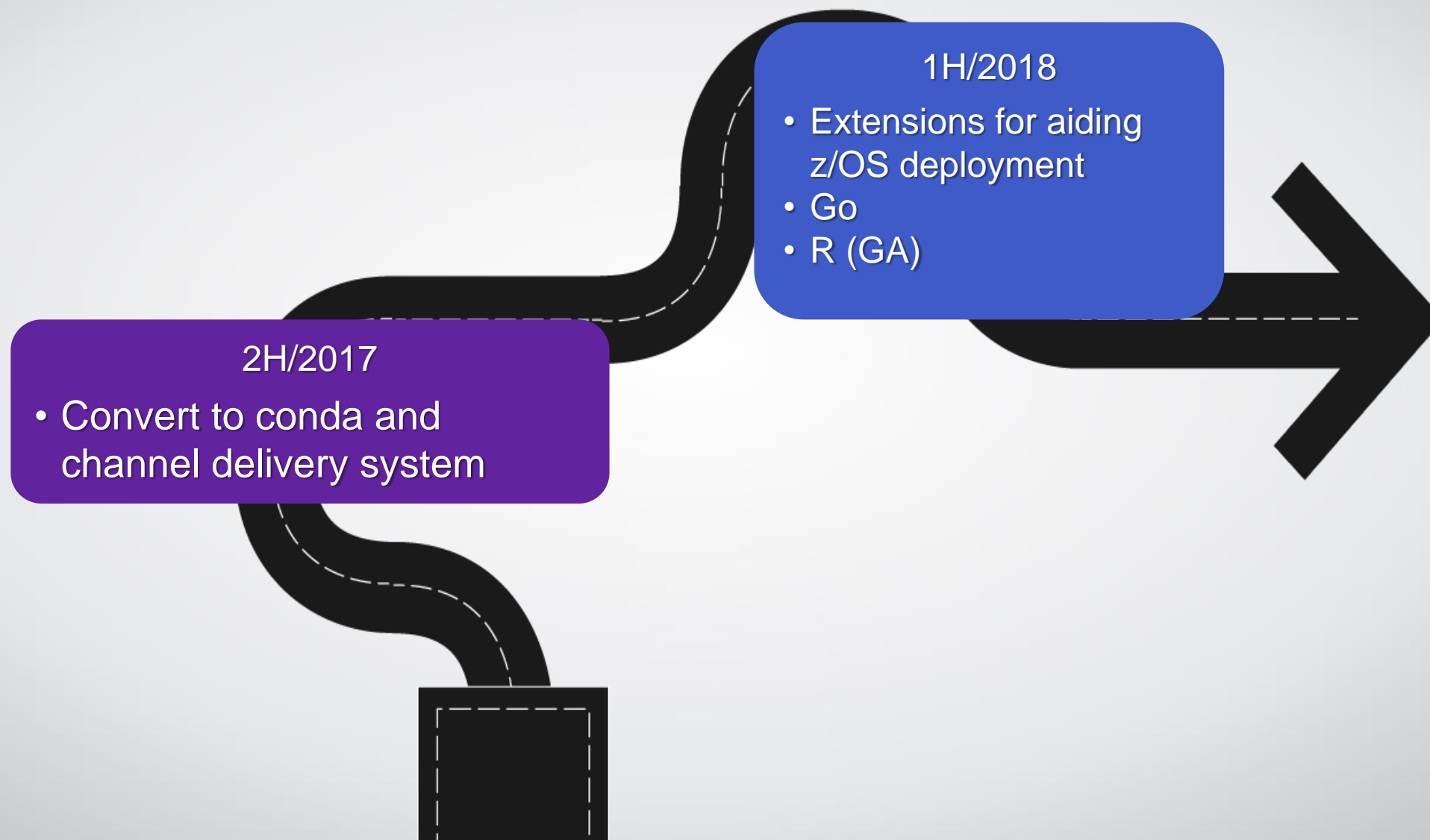
# Indices to access cols
IDX_ACTIVITY = 0
IDX_AGE = 1
IDX_INCOME = 2
IDX_INVEST = 3
IDX_CHURN = 4
IDX_ID = 5
IDX_GENDER = 6
IDX_EMAIL = 7
```

Also supported is Dask, a python-specific alternative for distributed processing

What can one do with this?



Roadmap – z/OS Open Source



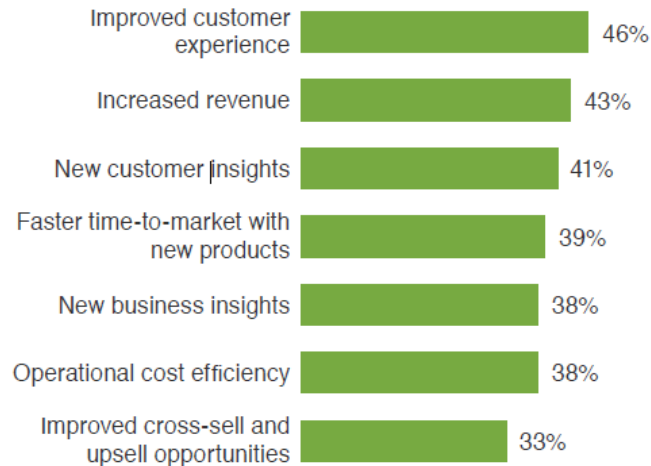
Data Analytics Survey: Need



FIGURE 8

Improved Analytics Drives Improved Customer Experience And Increases Revenue

“What are the business benefits you expect to receive through improving your data processing and analytical capabilities?”
(Select all that apply)



Base: 100 data science and data analytics leaders at enterprises within the US

Source: A commissioned study conducted by Forrester Consulting on behalf of IBM, May 2016

- Pervasiveness of Analytics
- Business growth

- Need for Real-Time Insight at point of impact

FIGURE 3

Nine Of 10 Data Scientists Are Interested In Real-Time Data For Modeling

“If there were no drawbacks (e.g., SLA concerns, resource consumption concerns) how interested would you be in having real-time data to use for modeling?”

91% of data scientists are interested/very interested in real-time data.

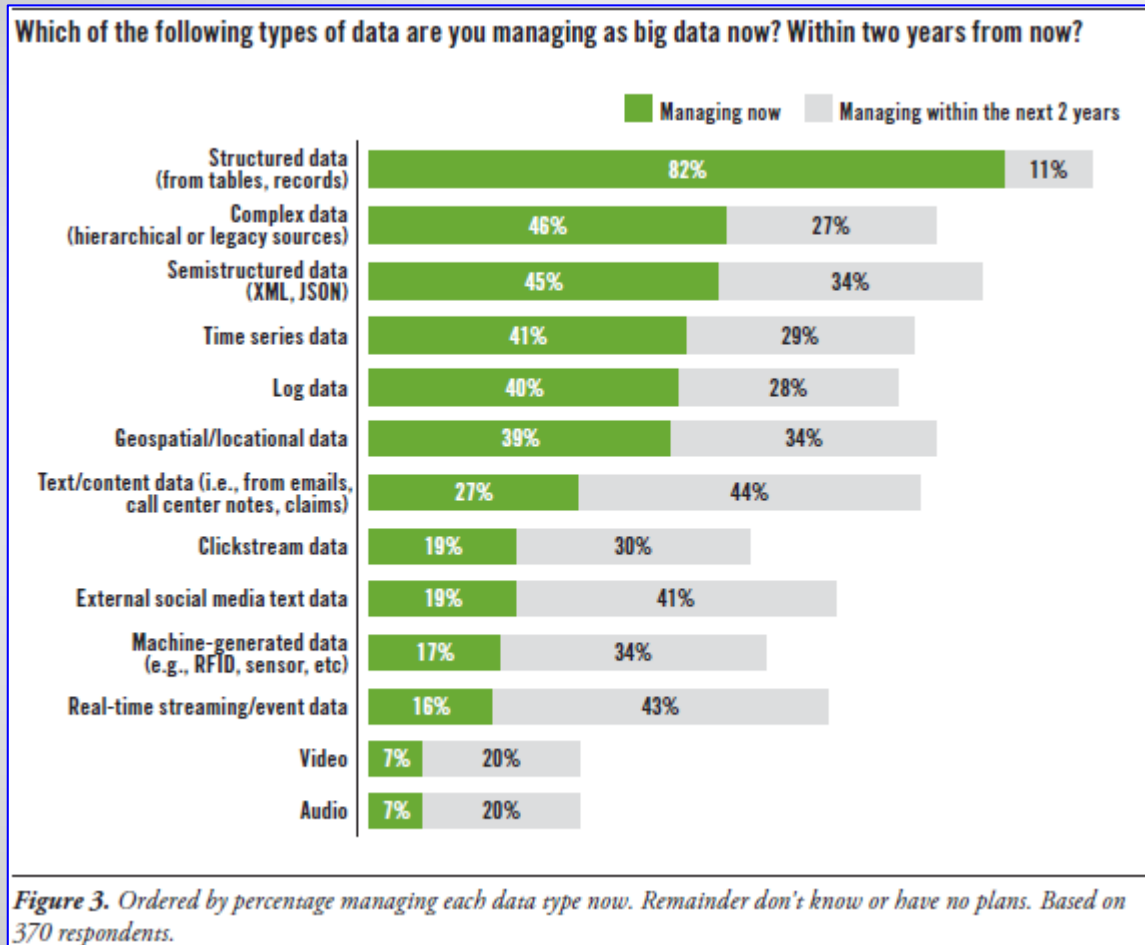


Base: 100 data science and data analytics leaders at enterprises within the US

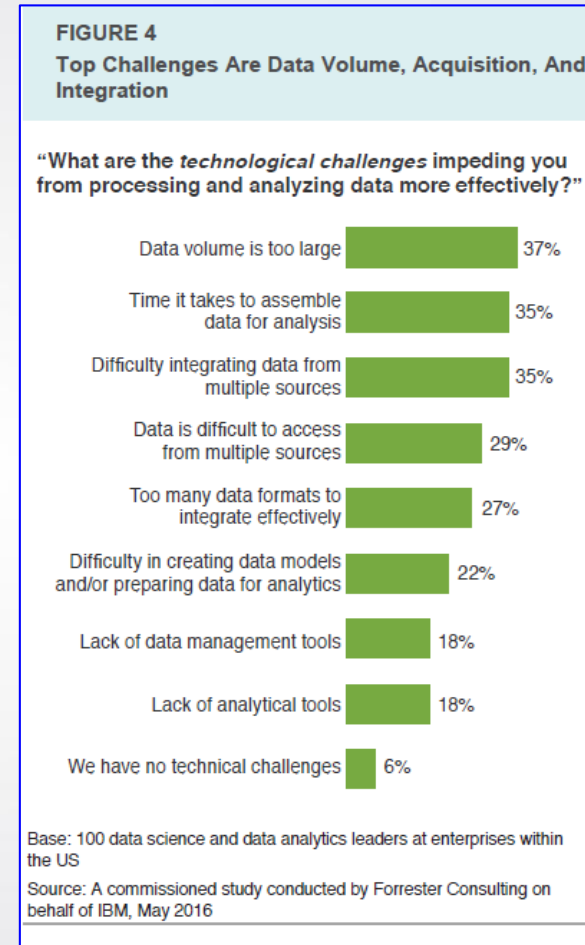
Source: A commissioned study conducted by Forrester Consulting on behalf of IBM, May 2016

Source & Full Forrester paper: <https://www-03.ibm.com/systems/z/solutions/real-time-analytics/data-analysis.html>

Data Analytics Survey: Challenges



Source: TDWI Best Practices Report, Dec 2016, Data Science and Big Data, by Fern Harper



Source: A commissioned study conducted by Forrester Consulting on behalf of IBM, May 2016

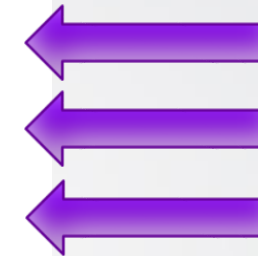
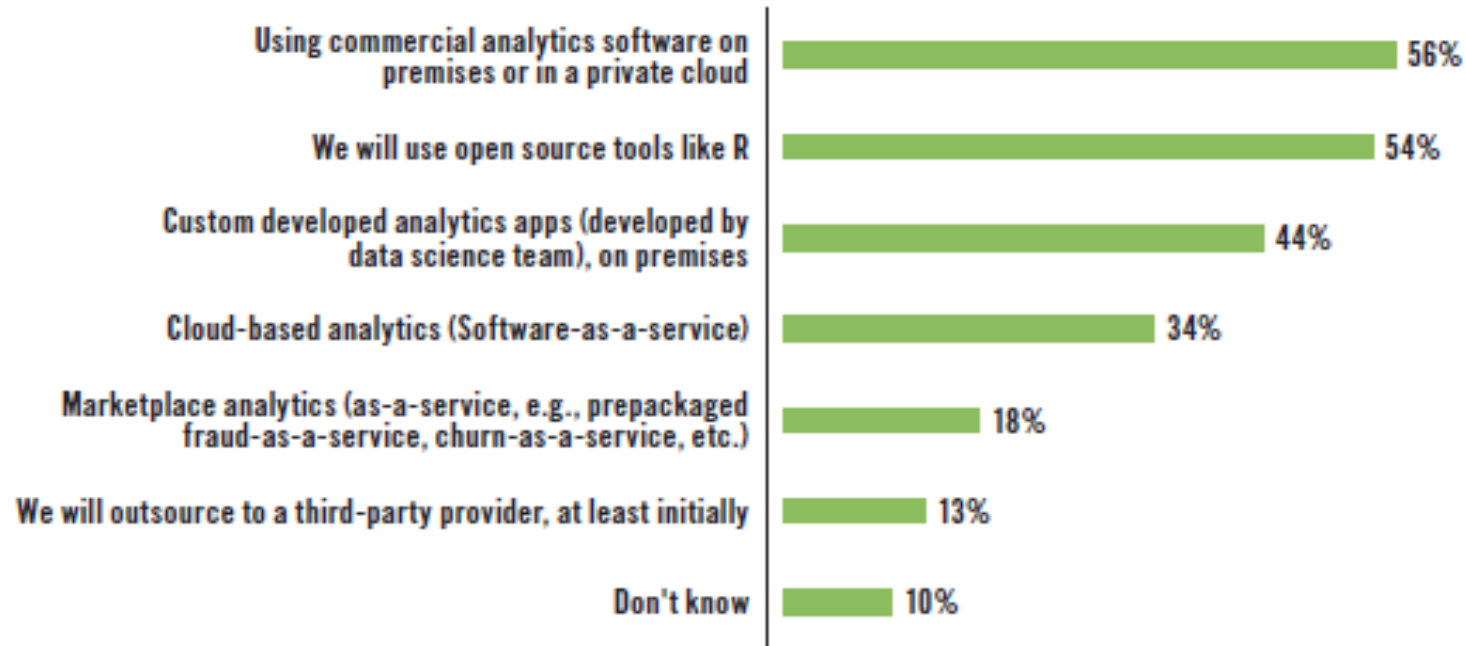
Challenges:

- Data integration
- Disparate sources
- Volume of Data
- Model creation and preparation

Data Analytics Survey: Methods



How is or do you believe your organization will deliver on your big data analytics projects? Please select all that apply.



On premise, leverage commercial analytics tools

Leverage open source

Custom applications, on premise

Figure 10. Based on 327 respondents.

Source: TDWI Best Practices Report, Dec 2016, Data Science and Big Data, by Fern Harper

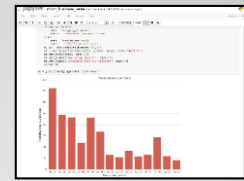
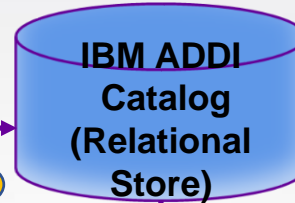
RDV data flow



Multitude of Enterprise Applications & Data



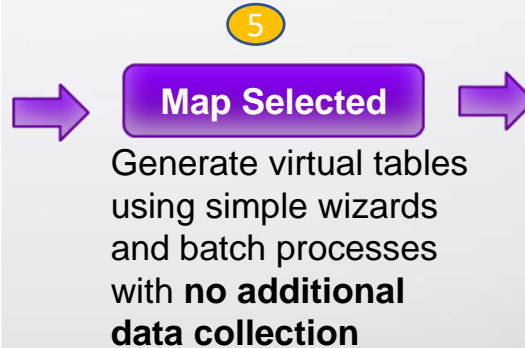
- Dataset names
- Cobol copybooks
- PL1 include files



Analytic Output

Match high level language structures to physical datasets and databases
Validate to associate type of data

	z/OS Dataset Name	Type	COBOL Structure
X	CSD.AI38.EMPLOYEE.KSDS	VSAM	EMPLOYEE-RECORD
	CSD.AI38.DEPARTMENT.KSDS	VSAM	DEPARTMENT-RECORD
X	CSD.AI38.OFFICE.KSDS	VSAM	OFFICE-RECORD
X	CSD.AI38.UPDATE.TRANS	QSAM	NAME-CHANGE-RECORD
X	CSD.AI38.UPDATE.TRANS	QSAM	ADDR-CHANGE-RECORD
	CSD.AI38.UPDATE.TRANS	QSAM	SALARY-CHANGE-RECORD
X	CSD.AI38.UPDATE.TRANS	QSAM	OFFICE-CHANGE-RECORD

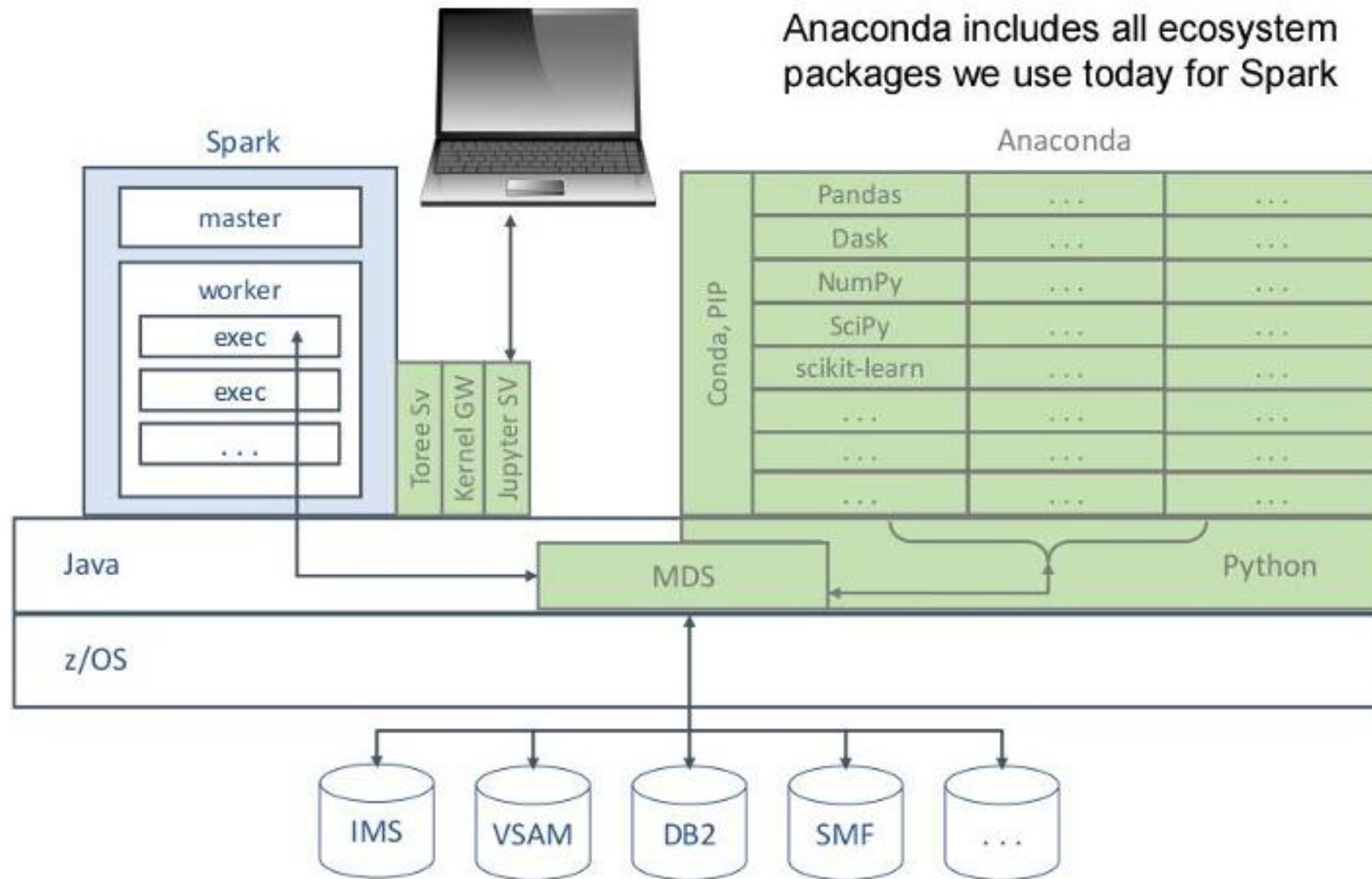


Cust_ID	Investment	Avg TX
1009530860	114368	2090.32
1009574010	112099	1297.41
1009578620	84638	1333.18

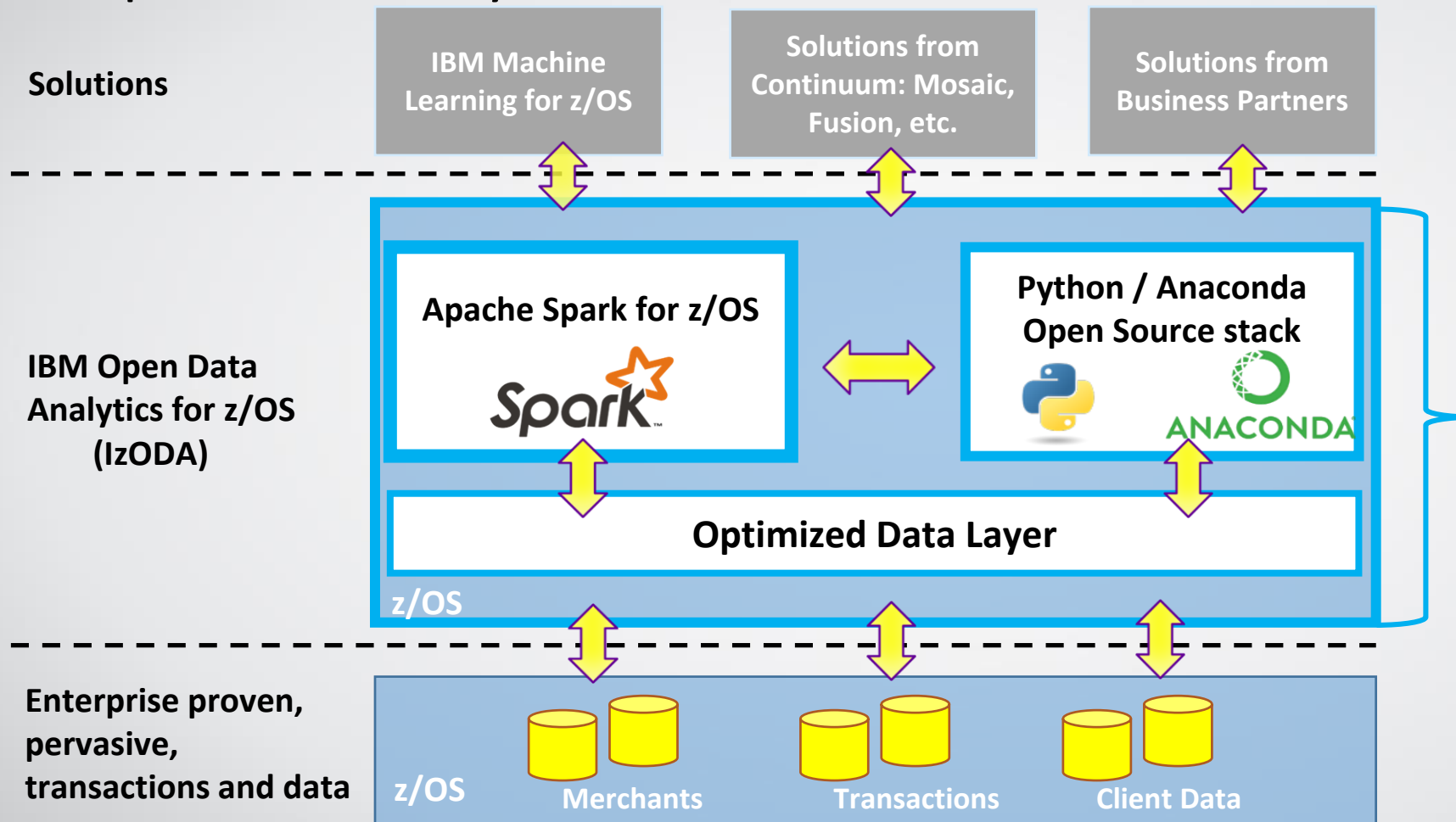
Virtual Tables Consumed by IBM Open Data Analytics for z/OS

66 | BELOW HERE, SURPLUS SLIDES





Applying Data-in-Place Analytics within the Enterprise: IBM Open Data Analytics for z/OS



- Establish rich runtimes for analytics co-located with IBM Z data
- Leverage leading open source technologies
- Enable advanced solutions from IBM and partners
- Integrate and differentiate with z/OS security & workload

Courtesy of Joe Bostian (jbostian@us.ibm.com), IBM Z Ecosystem Team Lead

IBM Open Data Analytics for z/OS V1.1



$$\text{IBM} + \text{Rocket} + \text{Continuum Analytics} = \text{https://izoda.github.io/}$$

Solutions

IBM Solutions

Partner Solutions

Optimized Analytics Platform

IBM Apache
Spark

Anaconda
for z/OS

Python

R

Optimized Data Access Layer (aka RDV)

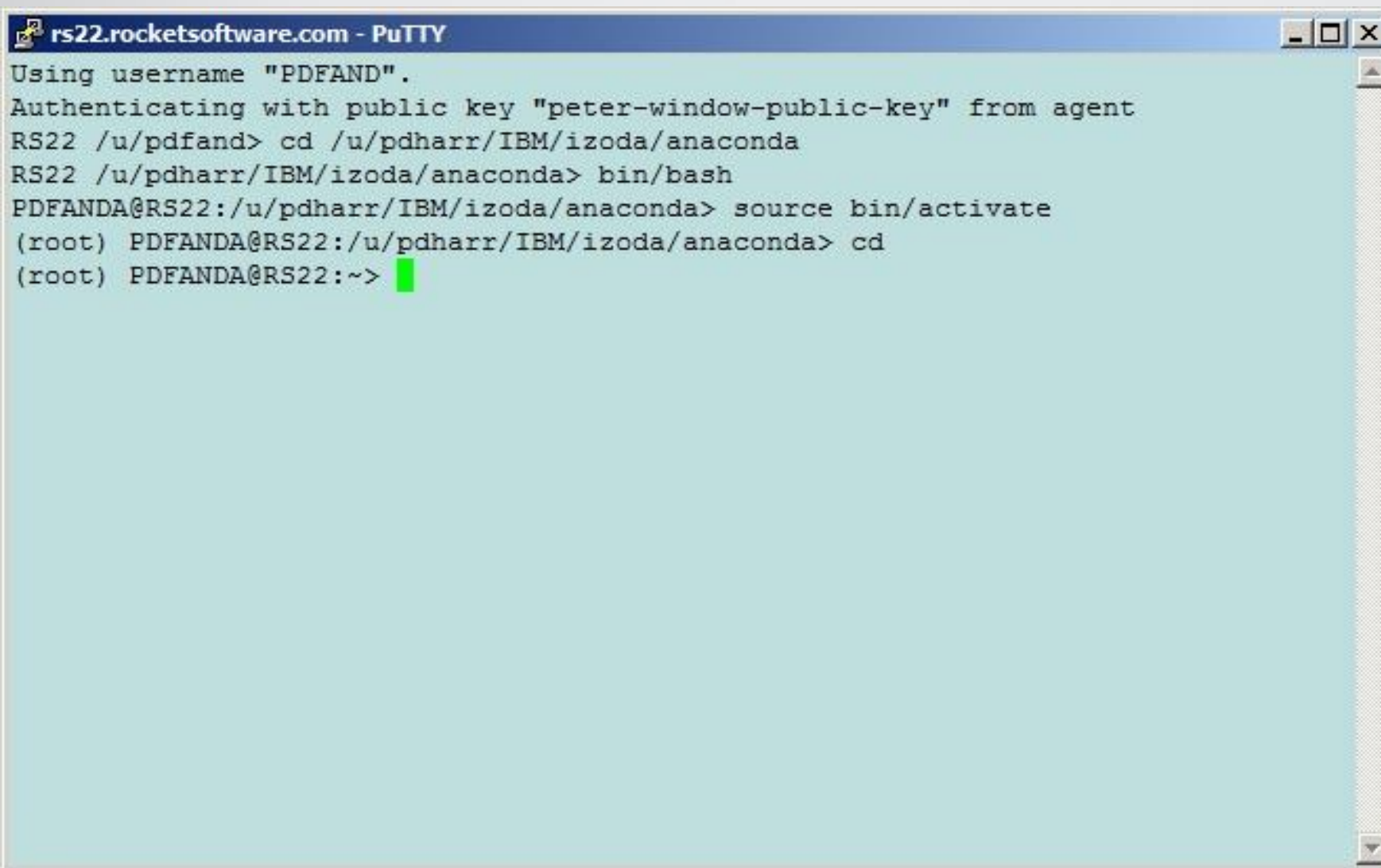
Your data!



A screenshot of a PuTTY terminal window titled "rs22.rocketsoftware.com - PuTTY". The window has a blue title bar and standard window controls (minimize, maximize, close) on the right. The terminal area has a light blue background. The text displayed is as follows:

```
Using username "PDFAND".  
Authenticating with public key "peter-window-public-key" from agent  
RS22 /u/pdfand> cd /u/pdharr/IBM/izoda/anaconda  
RS22 /u/pdharr/IBM/izoda/anaconda> █
```

A green cursor is visible at the end of the last line.



```
rs22.rocketsoftware.com - PuTTY
Using username "PDFFAND".
Authenticating with public key "peter-window-public-key" from agent
RS22 /u/pdfand> cd /u/pdharr/IBM/izoda/anaconda
RS22 /u/pdharr/IBM/izoda/anaconda> bin/bash
PDFANDA@RS22:/u/pdharr/IBM/izoda/anaconda> source bin/activate
(root) PDFANDA@RS22:/u/pdharr/IBM/izoda/anaconda> cd
(root) PDFANDA@RS22:~>
```

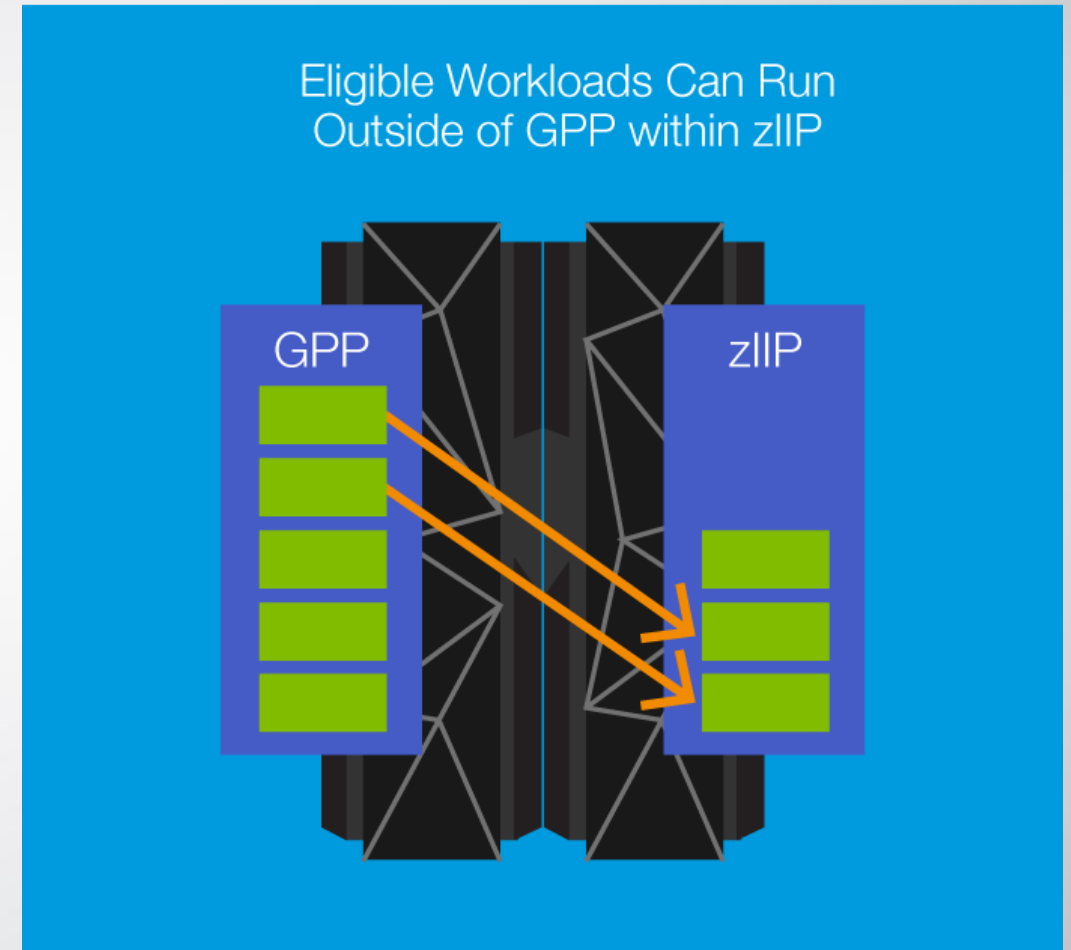

| Install latest Bash, git, and Perl



```
rs22.rocketsoftware.com - PuTTY
Using username "PDFAND".
Authenticating with public key "peter-window-public-key" from agent
RS22 /u/pdfand> cd /u/pdharr/IBM/izoda/anaconda
RS22 /u/pdharr/IBM/izoda/anaconda> bin/bash
PDFANDA@RS22:/u/pdharr/IBM/izoda/anaconda> source bin/activate
(root) PDFANDA@RS22:/u/pdharr/IBM/izoda/anaconda> cd
(root) PDFANDA@RS22:~> mkdir anthem
(root) PDFANDA@RS22:~> cd anthem
(root) PDFANDA@RS22:~/anthem> conda install bash=4.3 git perl
```

Addressing workload cost

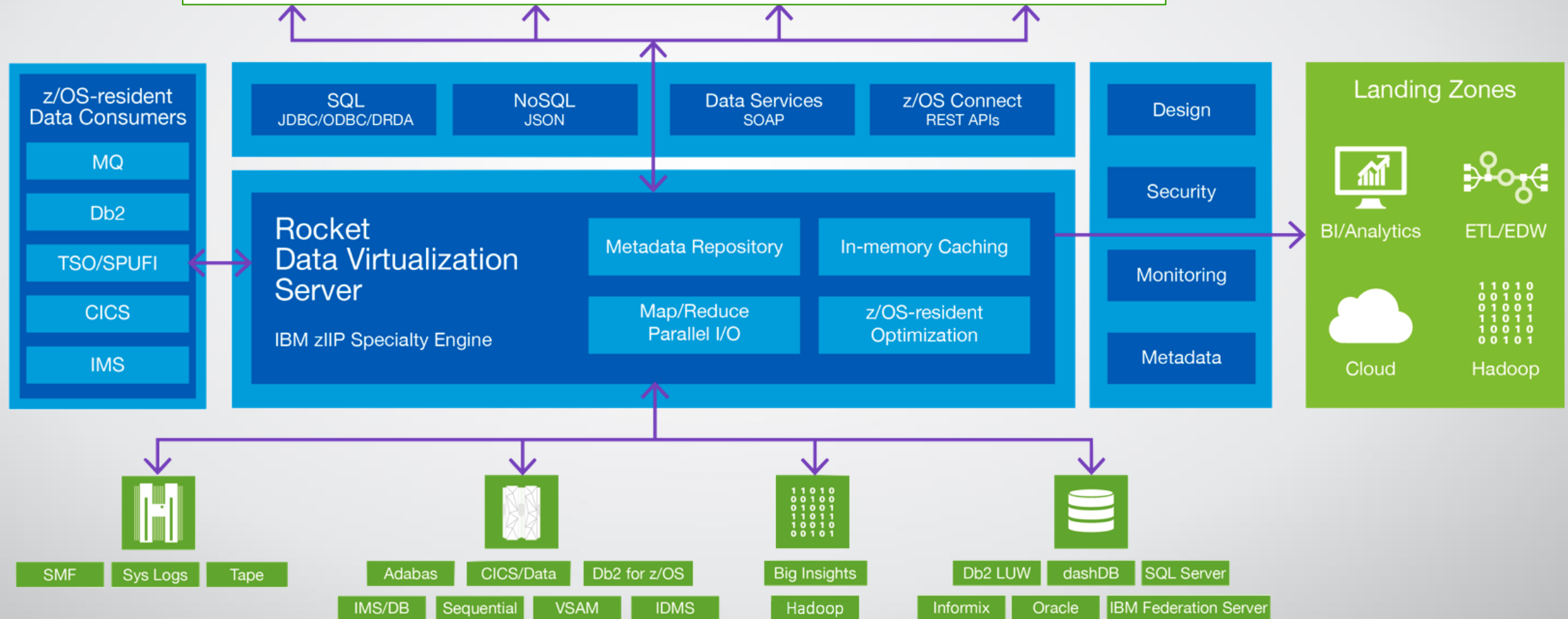
- Python and R (In IzODA)
 - zIIP-costing via workload metering
- RDV engine
 - Up to 99% of its own processing in the zIIP engine



Rocket Data Virtualization (RDV)



z/OS Open Source



Client Server Network Favorites

Host: rs28 (Port 1280)
Server: VDB1
Version: 02.01.00.0000
Set Server ...

SQL
Data
VDB1
Virtual Tables [U*]
US_ZIPCODES
ZIP [VARCHAR(5)]
CITY [VARCHAR(16)]
STATE [VARCHAR(2)]
POPULATION [DECIMAL(6,0)]
LOC_X [DECIMAL(9,6)]
LOC_Y [DECIMAL(8,6)]
Virtual Views
JDVT_VIEW
Other Subsystems
DBA8 [DB2Mbr]
DBA9 [DB2Mbr]
DCC1 [DB2Mbr]
DDS1 [DB2Grp]
DDS4 [DB2Grp]
DDS5 [DB2Grp]
DDS6 [DB2Grp]

Active Connections

DSN Name
rs28_VDB1

Common Tools

Generated.sql US_ZIPCODES

```
-- This statement will return all  
-- following table:  
-- Name      : US_ZIPCODES  
-- Catalog   : null  
-- Schema    : DVSQLE  
-- Remarks   : VSAM - DVS.VDB2.  
-- Tree Location: rs28/1280/SQL/Da  
-- The sql statement:  
SELECT ZIP, CITY, STATE, POPULATION  
FROM US_ZIPCODES LIMIT 100;
```

SQL Results Server Trace Console

	ZIP	CITY				
0	01001	AGAWAM				
1	01002	CUSHMAN				
2	01005	BARRE				
3	01007	BELCHERTOWN	MA	10579	72.410953	42.275103
4	01008	BLANDFORD	MA	1240	72.936114	42.182949
5	01010	BRIMFIELD	MA	3706	72.188455	42.116543
6	01011	CHESTER	MA	1688	72.988761	42.279421
7	01012	CHESTERFIELD	MA	177	72.833309	42.381670
8	01013	CHICOPEE	MA	23396	72.607962	42.162046
9	01020	CHICOPEE	MA	31495	72.576142	42.176443
10	01022	WESTOVER AFB	MA	1764	72.558657	42.196672

Columns Group: 1 of 1

Columns per group: 25

100 rows SQL Messages

New Virtual Table Wizard

Create a new VSAM Virtual Table

Name: LOGICAL_DATA_MART

Target: DVS.VDB1.USER.MAP

Description: Single View of Customer

☐ Convert VAR* fields to True VAR* fields

Arrays Handling

☒ Flatten arrays into a single fixed table at runtime (Y)

☐ Return arrays into separate tables at runtime (N)

☐ Flatten arrays now (C)

Current Server:

Host: rs28 Server: VDB1 (port 1280) Set Server...

? < Back Next > Finish Cancel