



DAT368 – Python Wrangling for SAP HANA Application Developers

PUBLIC

Speakers 2018



SAP TechEd Las Vegas

October 2–5, 2018

Andrew Lunde

Harini Gunabalan

SAP TechEd Barcelona

October 23–25, 2018

Aron MacDonald

Harini Gunabalan

SAP TechEd Bangalore

November 28–30, 2018

Abhijeet G Chinchanikar

Arun Srinivasan

Take the session survey.

We want to hear from you!

Be sure to complete the session evaluation
on the SAP TechEd mobile app.



Download the app from
iPhone App Store or Google Play.



Disclaimer

The information in this presentation is confidential and proprietary to SAP and may not be disclosed without the permission of SAP. Except for your obligation to protect confidential information, this presentation is not subject to your license agreement or any other service or subscription agreement with SAP. SAP has no obligation to pursue any course of business outlined in this presentation or any related document, or to develop or release any functionality mentioned therein.

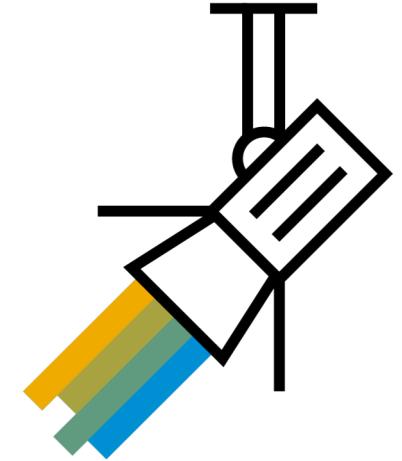
This presentation, or any related document and SAP's strategy and possible future developments, products and or platforms directions and functionality are all subject to change and may be changed by SAP at any time for any reason without notice. The information in this presentation is not a commitment, promise or legal obligation to deliver any material, code or functionality. This presentation is provided without a warranty of any kind, either express or implied, including but not limited to, the implied warranties of merchantability, fitness for a particular purpose, or non-infringement. This presentation is for informational purposes and may not be incorporated into a contract. SAP assumes no responsibility for errors or omissions in this presentation, except if such damages were caused by SAP's intentional or gross negligence.

All forward-looking statements are subject to various risks and uncertainties that could cause actual results to differ materially from expectations. Readers are cautioned not to place undue reliance on these forward-looking statements, which speak only as of their dates, and they should not be relied upon in making purchasing decisions.

Agenda

Why are we here?

- Why Python?
- Why microservices?
- Why MTA?
- Why on prem (or Cloud)?



Building Blocks

- Module Assembly, Staging and Execution + Dependency Resolution
- Buildpack Assembly Process

Exercises

- MTA Project – Web IDE Usage + XS CLI
- Building and Deploying MTAR
- Compilation and Installation of Python Runtime
- Building and Deploying to the Cloud

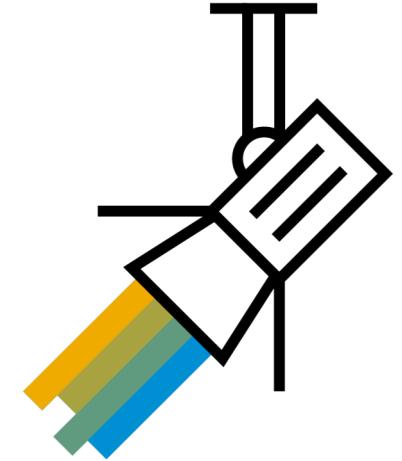
“Always code as if the guy who ends up maintaining your code will be a violent psychopath who knows where you live.”

Martin Golding

Agenda

Why are we here?

- Why Python?
- Why microservices?
- Why MTA?
- Why on prem (or Cloud)?



Building Blocks

- Module Assembly, Staging and Execution + Dependency Resolution
- Buildpack Assembly Process

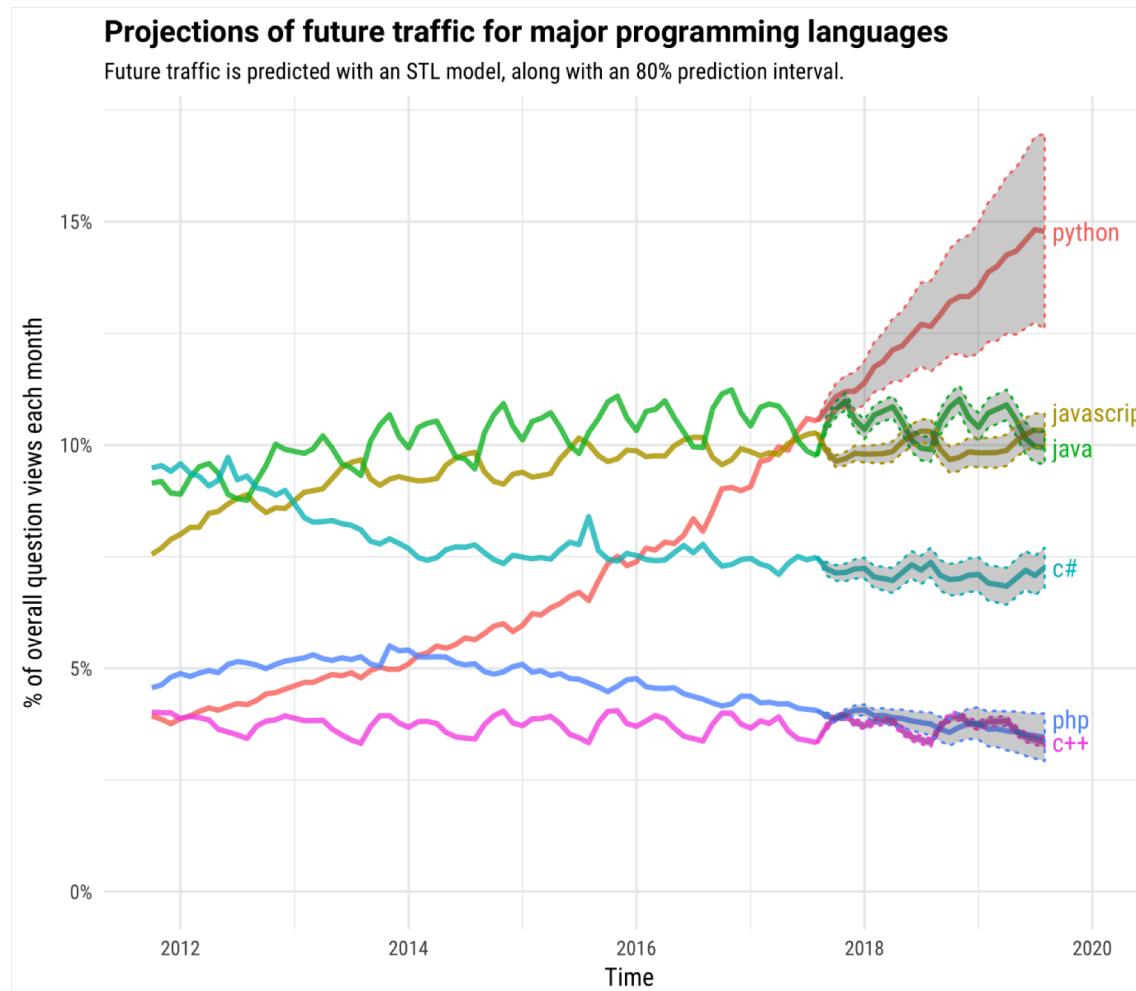
Exercises

- MTA Project – Web IDE Usage + XS CLI
- Building and Deploying MTAR
- Compilation and Installation of Python Runtime
- Building and Deploying to the Cloud

Why Python?

All the rage (at least for machine learning)

Python is expected to grow faster for machine learning



<https://medium.com/@UdacityINDIA/why-use-python-for-machine-learning-e4b0b4457a77>

Why Python?

All the rage (at least for machine learning)

Existing Code (porting destroys prior confidence)

Appropriateness for Task

- Want to work with images—numpy, opencv, scikit
- Want to work in text—nltk, numpy, scikit
- Want to work in audio—librosa
- Want to solve machine learning problem—pandas, scikit
- Want to see the data clearly—matplotlib, seaborn, scikit
- Want to use deep learning—tensorflow, pytorch
- Want to do scientific computing—scipy
- Want to integrate web applications—Django

Why microservices?

There is only one codebase per app(module), but there will be many deploys of the app. A deploy is a running instance of the app. <https://12factor.net/codebase>

- Follow the Single Responsibility Principle (does one task well)
- Resilient/Flexible – failure in one service does not impact other services
- High scalability – demanding services can be deployed in multiple servers to enhance performance and keep away from other services so that they don't impact other services.
- Easy to enhance – less dependency and easy to change and test
- Low impact on other services – being an independent service, this has less chance to impact other services
- Easy to understand since they represent the small piece of functionality
- Ease of deployment (also flexible to deploy in different environments like cloud or on premis)
- Freedom to choose technology – allows you to choose language that is best suited for a particular functionality (Python anyone?)

Why (what is) an MTA?

SAP has built upon microservices the concept of a Multi-Target Application or MTA.

The MTA specification addresses this life-cycle and orchestration complexity of continuous deployment for cloud and on-premise platforms. We use the following definition:

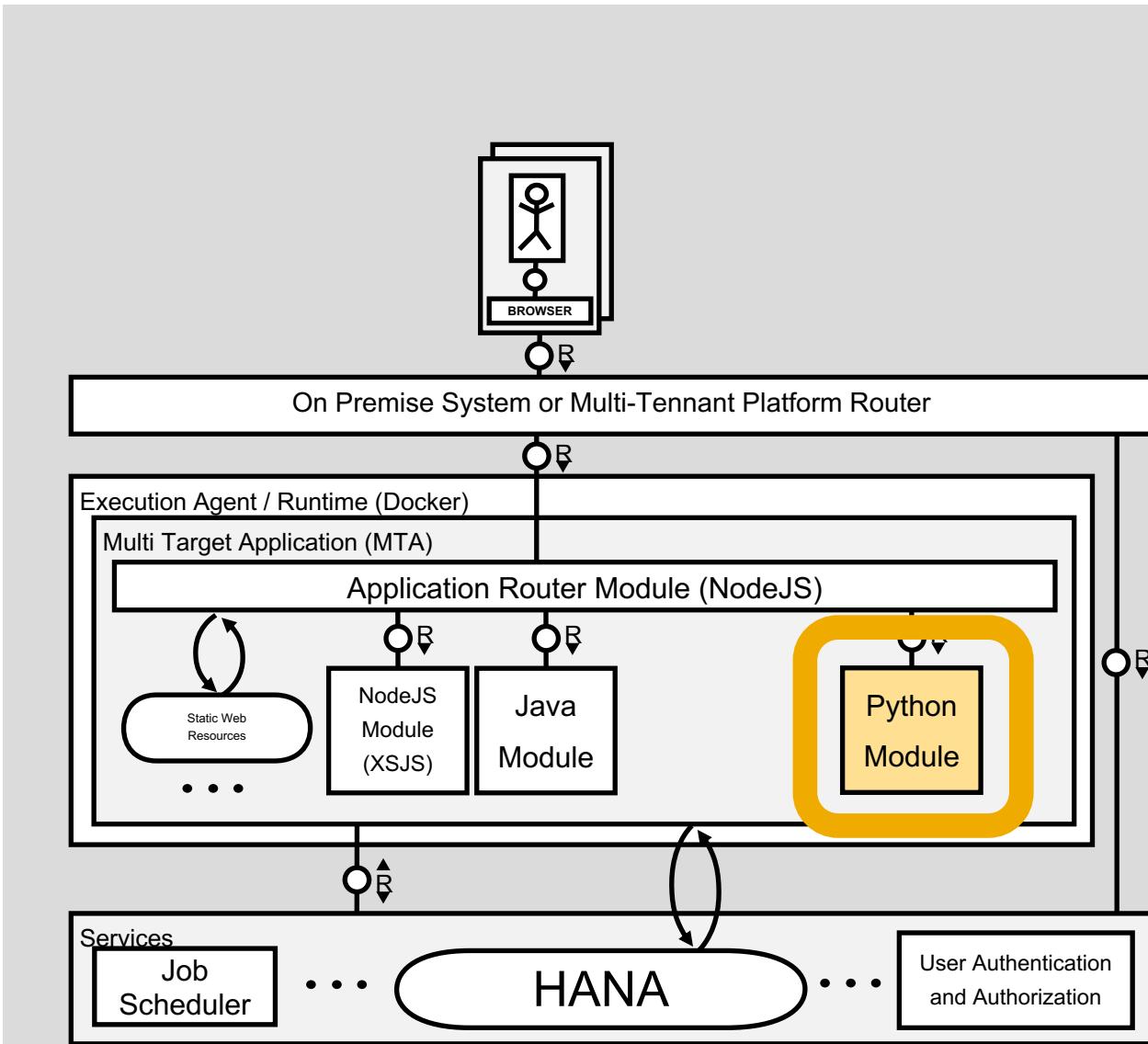
*A multi-target application (MTA) is comprised of multiple parts (“modules”), created with different technologies and deployed to different targets, but with a **single, common lifecycle**.*

The MTA addresses the deployment challenges by isolating the developer from target-specific native tools (like XS push), via a formal target-independent **application model**. Developers describe the modules of the application, the interdependencies on other modules, MTAs and (micro) services, and required and exposed interfaces. An MTA-aware application lifecycle management framework validates, orchestrates and automates the MTA deployment on premise and on cloud platforms.

Why (what is) an MTA? (continued)

Isolation is provided by run-time

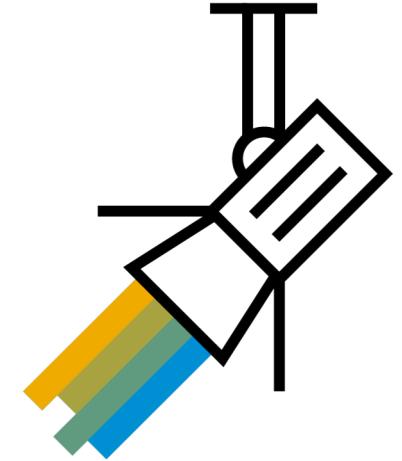
- Combined with a specific buildpack that completes the execution environment
- Can be started/stopped/scaled
- Does work appropriate to language being used to implement it (in our case, python)



Agenda

Why are we here?

- Why Python?
- Why microservices?
- Why MTA?
- Why on prem (or Cloud)?



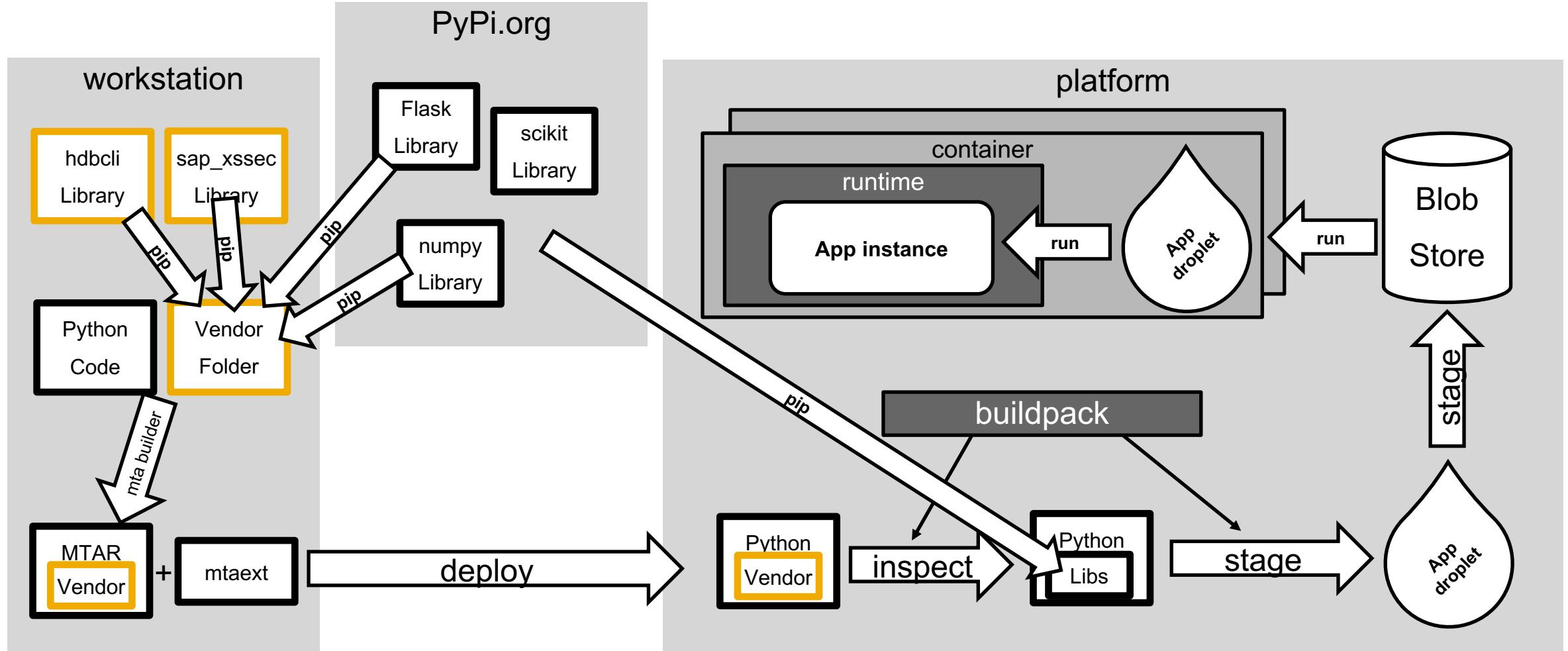
Building Blocks

- Module Assembly, Staging and Execution + Dependency Resolution
- Buildpack Assembly Process

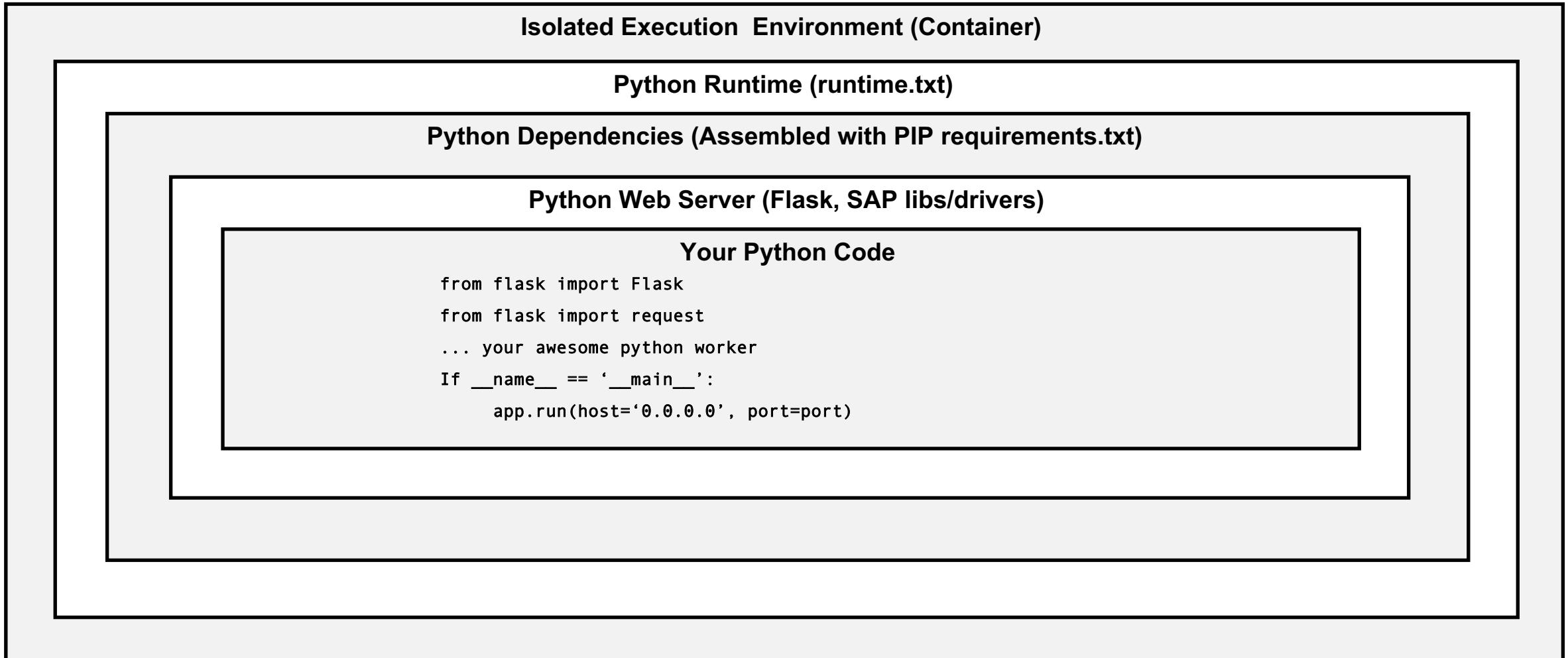
Exercises

- MTA Project – Web IDE Usage + XS CLI
- Building and Deploying MTAR
- Compilation and Installation of Python Runtime
- Building and Deploying to the Cloud

Module Assembly, Staging and Execution + Dependency Resolution



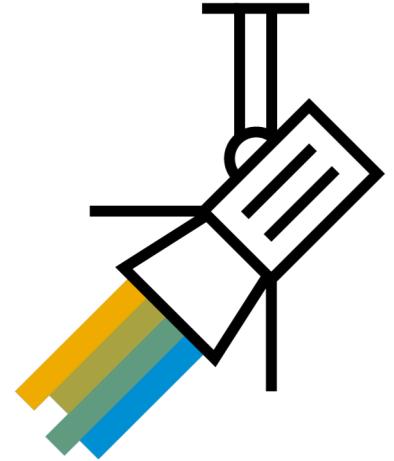
Buildpack Assembly Process



Agenda

Why are we here?

- Why Python?
- Why microservices?
- Why MTA?
- Why on prem (or Cloud)?



Building Blocks

- Module Assembly, Staging and Execution + Dependency Resolution
- Buildpack Assembly Process

Exercises

- MTA Project – Web IDE Usage + XS CLI
- Building and Deploying MTAR
- Compilation and Installation of Python Runtime
- Building and Deploying to the Cloud

Python Workshop Exercises

Web IDE for SAP HANA

Putty SSH Client + XS CLI

Role Collection Setup

MTA Rep Build

MTAR Deploy (no Python)

Compile and install Python

Enable SAP Python Libs

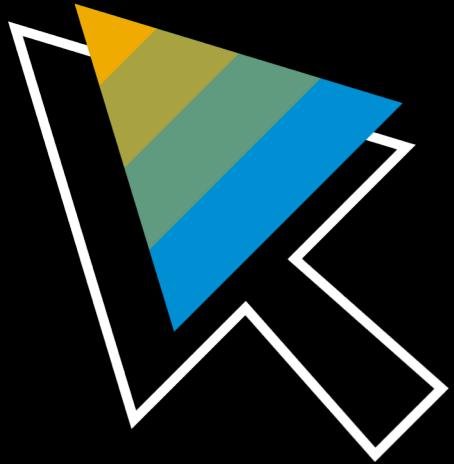
MTA Build with Python

MTAR Deploy with Python

Deploy Python MTAR to Cloud Foundry



Exercise



Watch for the Time Check notes and skip ahead to save time.

After some initial setup steps you should be spending most of your time in Exercise 3.

SAP TechEd online / SAP Community

Access replays of:

- Keynotes
- SAP TechEd live interviews
- Selected lecture sessions

<http://sapteched.com/online>

Continue your SAP TechEd discussion after the event within SAP Community:

- Read and reply to blog posts
- Ask your questions
- Join conversations

<sap.com/community>

See all [SAP TechEd blog posts](#)

The screenshot shows the SAP TechEd Online homepage. At the top, there's a navigation bar with links for Home, 2017 Videos, Archive, Blogs, and SAP TechEd. On the right, there's a user icon and a yellow 'Registration' button. The main banner features the text 'SAP TechEd Online' and 'Registration for Las Vegas is Open Now' with a 'Get Registration Information' link. Below the banner is a graphic of a large audience in a conference hall. To the right of the audience, there's a 'Contact us' button. A 'Featured Blog' section is visible, showing a profile picture of a woman and a link to a blog post about SAP TechEd in 2018.

SAP TechEd Online
Registration for Las Vegas is Open Now
[Get Registration Information](#)

Featured Blog

 SAP TechEd in 2018:
Register for this Year's Ultimate How-to Learning Adventure

SAP TechEd in 2018 will give attendees the personalized, inspirational learning journey of their careers. I'll be video reporting from the show floor, and hope to see you there. Read more on [my blog](#).

[Go To Blog](#)

Further information



Related SAP TechEd content journeys

- DAT3 – Build Applications Using SAP HANA Native Development
 - OPP2 – Build Cloud Native Apps with SAP Cloud Platform
 - EXP3 – Level Up as a Developer
-

Related SAP TechEd sessions

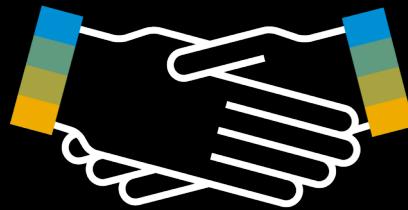
- DAT101 – What's New in Native Development for SAP HANA
 - DAT201 – Developing Applications Using Predictive and Machine Learning in SAP HANA
 - DAT364 – Developing Smart Applications Using SAP HANA In-Database Machine Learning
 - DAT815 – Machine Learning in SAP HANA
 - OPP621 – Choose and Use the Right Programming Languages with SAP Cloud Platform
-

Public SAP Web sites

- SAP Community: www.sap.com/community
 - SAP products: www.sap.com/products
-

SAP training and certification opportunities

www.sap.com/education



Thanks for attending this session.



Feedback

Please complete your session evaluation for **DAT368**.

Contact for further topic inquiries

Andrew Lunde
Principal Solution Architect
andrew.lunde@sap.com