The Road Ahead

Morten Kromberg, CXO, Dyalog
Time to find new maps!

Oops, No I meant the First

You’ll have to go round again
(M32, M4)
(M32)

Take the second exit

Look I said I was Sorry

The Road Ahead
Time to find new maps!
Time to find new maps!
Glimpses of a Modern User Experience

Imagine you meet a young data scientist (let's call her Mary) who says...
Glimpses of a Modern User Experience

Imagine you meet a young data scientist (let's call her Mary) who says...

*My professor says that you know something about this cool new language for analytics - called "APL"?*
Glimpses of a Modern User Experience

Imagine you meet a young data scientist (let's call her Mary) who says...

*My professor says that you know something about this cool new language for analytics - called "APL"?*

*I have this crazy idea that it would be nice to count the frequency of digits used in numeric fields within CSV files, to check for fake data(!)*
Example...

<table>
<thead>
<tr>
<th>Type</th>
<th>North</th>
<th>South</th>
<th>East</th>
<th>West</th>
</tr>
</thead>
<tbody>
<tr>
<td>Red</td>
<td>123</td>
<td>270</td>
<td>377</td>
<td>187</td>
</tr>
<tr>
<td>Blue</td>
<td>357</td>
<td>377</td>
<td>124</td>
<td>179</td>
</tr>
</tbody>
</table>
Example...

<table>
<thead>
<tr>
<th>Type</th>
<th>North</th>
<th>South</th>
<th>East</th>
<th>West</th>
</tr>
</thead>
<tbody>
<tr>
<td>Red</td>
<td>123</td>
<td>270</td>
<td>377</td>
<td>187</td>
</tr>
<tr>
<td>Blue</td>
<td>357</td>
<td>377</td>
<td>124</td>
<td>179</td>
</tr>
</tbody>
</table>
**Example...**

<table>
<thead>
<tr>
<th>Type</th>
<th>North</th>
<th>South</th>
<th>East</th>
<th>West</th>
</tr>
</thead>
<tbody>
<tr>
<td>Red</td>
<td>123</td>
<td>270</td>
<td>377</td>
<td>187</td>
</tr>
<tr>
<td>Blue</td>
<td>357</td>
<td>377</td>
<td>124</td>
<td>179</td>
</tr>
</tbody>
</table>

Digit Counts (ignoring 1\textsuperscript{st})

```
0 1 2 3 4 5 6 7 8 9
1 0 1 1 1 1 1 8 1 1
```
**Example...**

<table>
<thead>
<tr>
<th>Type, North, South, East, West</th>
</tr>
</thead>
<tbody>
<tr>
<td>Red, 123, 270, 377, 187</td>
</tr>
<tr>
<td>Blue, 357, 377, 124, 179</td>
</tr>
</tbody>
</table>

Digit Counts (ignoring 1st)

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>8</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

FAKE DATA ALERT!
Example...

<table>
<thead>
<tr>
<th>Type</th>
<th>North</th>
<th>South</th>
<th>East</th>
<th>West</th>
</tr>
</thead>
<tbody>
<tr>
<td>Red</td>
<td>123</td>
<td>270</td>
<td>377</td>
<td>187</td>
</tr>
<tr>
<td>Blue</td>
<td>357</td>
<td>377</td>
<td>124</td>
<td>179</td>
</tr>
</tbody>
</table>

Digit Counts (ignoring 1st)

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>8</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

... but I can't find a Python Library to do this 😐
Example...

<table>
<thead>
<tr>
<th>Type</th>
<th>North</th>
<th>South</th>
<th>East</th>
<th>West</th>
</tr>
</thead>
<tbody>
<tr>
<td>Red</td>
<td>123</td>
<td>270</td>
<td>377</td>
<td>187</td>
</tr>
<tr>
<td>Blue</td>
<td>357</td>
<td>377</td>
<td>124</td>
<td>179</td>
</tr>
</tbody>
</table>

Digit Counts (ignoring 1st)

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>8</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

... but I can't find a Python Library to do this 😞
... can you help? 😊
You came to the right place ...
You came to the right place ...
You came to the right place ...
Preparing Mary's Linux Machine
Preparing Mary's Linux Machine

• Install git
Preparing Mary's Linux Machine

• Install git

```bash
yum install git
```
Preparing Mary's Linux Machine

- Install git
  - `yum install git`
- Install docker
Preparing Mary's Linux Machine

- Install git
- Install docker

```
yum install git
yum install -y docker
usermod -a -G docker mary
```
Preparing Mary's Linux Machine

- Install git
- Install docker
- Install Dyalog APL

```
yum install git
yum install -y docker
usermod -a -G docker mary
```
Preparing Mary's Linux Machine

- Install git
- Install docker
- Install Dyalog APL

```
yum install git
yum install -y docker
usermod -a -G docker mary
apt-get install dyalog-unicode
```
Preparing Mary's Linux Machine

- Install git
- Install docker
- Install Dyalog APL

```
yum install git
yum install -y docker
usermod -a -G docker mary
```

```
apt-get install dyalog-unicode
```
Preparing Mary's Linux Machine

- Install git
- Install docker
- Install Dyalog APL
- Grab Dyalog's Docker Utils

```bash
yum install -y docker
usermod -a -G docker mary
apt-get install dyalog-unicode
```
Preparing Mary's Linux Machine

- Install git
- Install docker
- Install Dyalog APL
- Grab Dyalog's Docker Utils

```
yum install git
yum install -y docker
usermod -a -G docker mary
apt-get install dyalog-unicode
```

```
git clone https://github.com/dyalog/docker-utils
```
Preparing Mary's Linux Machine

- Install git
- Install docker
- Install Dyalog APL
- Grab Dyalog's Docker Utils
- Clone our noodlings

```bash
yum install git
yum install -y docker
usermod -a -G docker mary
apt-get install dyalog-unicode

```

```bash
git clone https://github.com/dyalog/docker-utils
```

```bash
``
Preparing Mary's Linux Machine

• Install git
• Install docker
• Install Dyalog APL
• Grab Dyalog's Docker Utils
  - yum install git
  - yum install -y docker
  - usermod -a -G docker mary
  - apt-get install dyalog-unicode
  - git clone https://github.com/dyalog/docker-utils
  - git clone https://github.com/mkromberg/d18mary
• Clone our noodlings
  - git clone https://github.com/mkromberg/d18mary/d18demo

#dyalog18

The Road Ahead
Continuing under Linux ...
### In order of appearance...

<table>
<thead>
<tr>
<th>Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>APL</strong></td>
</tr>
<tr>
<td><strong>GitHub</strong></td>
</tr>
<tr>
<td><strong>Docker &amp; DockerHub</strong></td>
</tr>
<tr>
<td><strong>Visual Studio Code</strong></td>
</tr>
<tr>
<td><strong>EC2</strong></td>
</tr>
</tbody>
</table>
The Cloud is a Good Home for APL
The Cloud is a Good Home for APL

- The clouds favour lightweight, compact tools that do not need big frameworks
The Cloud is a Good Home for APL

- The clouds favour lightweight, compact tools that do not need big frameworks
- Simple APIs
The Cloud is a Good Home for APL

- The clouds favour lightweight, compact tools that do not need big frameworks
- Simple APIs
- APL is coming home
The Cloud is a Good Home for APL

- The clouds favour lightweight, compact tools that do not need big frameworks
- Simple APIs
- APL is coming home
All you need is ... Docker

• As soon as Docker is installed, the rest is easy

• Packaging, distribution and scaling is simple
  o Applications – WITH dependencies - can be up and running on any platform in seconds
  o (and the "reverse": sending documentation of problems back to developers is now so easy!)

• Equally attractive to
  o New users who wanted to get started quickly
  o Corporations who need to
    ▪ deploy applications on the cloud, or
    ▪ implement "Continuous Integration" workflows
Work for Dyalog To Do
Work for Dyalog To Do

• New licences to enable public installers and containers:
  ○ apt install dyalog
  ○ docker pull dyalog:17.1
Work for Dyalog To Do

• New licences to enable public installers and containers:
  o apt install dyalog
  o docker pull dyalog:17.1

• More public containers and cloud images (dyalog, jsonserver, miserver, jupyter, tamstat, ...)

#dyalog18
Work for Dyalog To Do

• New licences to enable public installers and containers:
  o apt install dyalog
  o docker pull dyalog:17.1

• More public containers and cloud images (dyalog, jsonserver, miserver, jupyter, tamstat, ...)

• Integration with VS Code, Emacs and other tools
Work for Dyalog To Do

• New licences to enable public installers and containers:
  o apt install dyalog
  o docker pull dyalog:17.1

• More public containers and cloud images (dyalog, jsonserver, miserver, jupyter, tamstat, ...)

• Integration with VS Code, Emacs and other tools

• "Project Model"
  o Define a "Dyalog APL Project" structure
  o Dependency management
  o Unit (& other) Testing
Work To Do
Work To Do

• A bridge from APL to the dotnet core
  o Access to rapidly growing collection of cross-platform utilities and libraries
Work To Do

• A bridge from APL to the dotnet core
  o Access to rapidly growing collection of cross-platform utilities and libraries

• Support pure script-based applications
  o Notation for Array Constants (script-able data)
  o "#!" script support
New Maps and Signs ...
Related talks...

Today:
- D05: RIDE 4.1 and Next Generation Integrations  (Gilgamesh Athoraya)
- D04: Array Notation Mk III                  (Adam Brudzewsky)
- D06: Cross-Platform User Interfaces         (Brian Becker)

Tomorrow:
- U06: The Workspace is Dead! Long Live the Workspace! (Paul Mansour)
- U05: The APL Package Manager                 (Gil)

Wednesday:
- U09: The evolution of the APL Tree Library   (Kai Jaeger)
- D11: Cloud Computing with APL               (Morten)
- U15: ⎕WC on the Web                         (Chris & Michael Hughes)
- U16: Serverless APL                         (Marko Vranic)

Thursday:
- D12: Jupyter Notebooks                      (Adam)
Tune in again next year...
Tune in again next year...