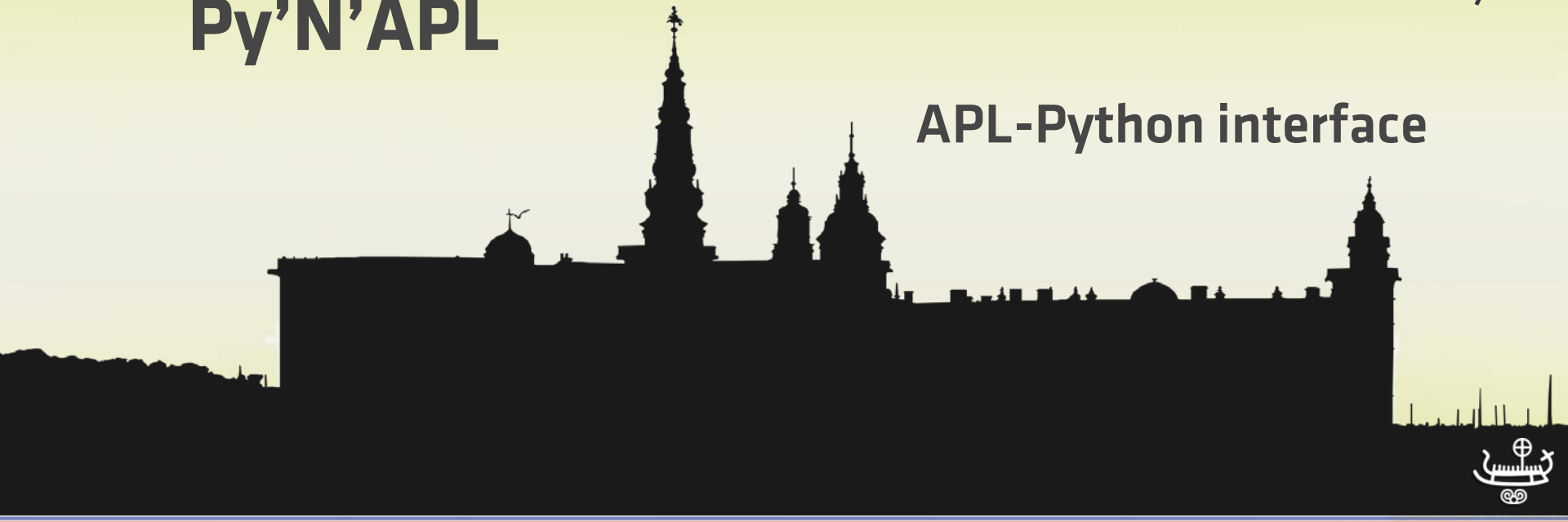
DYALOG

Elsinore 2017

Py'N'APL

APL-Python interface



Why interface with Python?

- Highly popular scripting language
- Dynamically typed
- Can be used interactively
- Available on Linux, Windows, Mac, RPi
- Used by hobbyists, scientists, and businesses
- Many libraries available

Why interface with Python?

- Cross-platform GUI; language processing; image processing; robotics; networking; social media
- There's even an add-on for multidimensional arrays
- Also used as an embedded scripting language in e.g. Blender
- If it exists, there's a Python library

```
>>> def fib(n):
        a, b = 0, 1
        while a<n:
                print(a, end=' ')
                a, b = b, a+b
        print()


>>> fib(1000)
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987
```

Py'n'APL

- Allow APL to start up and control a Python interpreter, and vice versa
- Exchange data between the two interpreters
- Call functions defined in one language from the other

Data conversion

Python	APL
int, float, complex, long (2)	number
list, tuple, iterable	vector
dictionary	namespace
length-1 string	character
length>1 string	character vector

Demo

Finally

- Requirements:
  - Dyalog 16 Unicode
  - Python 2.7.9 or higher; Python 3.4 or higher.
- Code is on GitHub:
  - https://github.com/Dyalog/pynapl
- Still very experimental