

# Making your own Firebird PowerTool (with Python)

Pavel Císař  
IBPhoenix



Are you insane?  
There is already a lot of  
quite powerfull tools to  
pick from, so why on  
Earth we should want to  
make our own?



# Why? Because all these tools...

- ✓ Are hard to use, or...
- ✓ Haven't feature(s) you want, or...
- ✓ Are not flexible enough, or...
- ✓ Are hard to learn, or...
- ✓ Doesn't work on your platform, or...
- ✓ Scalle badly, or...
- ✓ Disappoint you in any oter way.



# Inherent characteristics of existing tools

## CLI tools

- + Very Flexible, multiplatform
- Poor presentation environment, not powerful enough (ISQL) or powerful, but outdated (QLI), hard to use or learn

## GUI tools

- + Rich presentation environment, good for in-place changes and common tasks
- Scale badly, WYSIWYG plague, Not very flexible, mostly Windows-only

## Web-based tools

- + Multiplatform
- "The worst from both worlds"



# PowerTool™ must...

- ✓ Provide easy and obvious way to do frequent tasks
- ✓ Don't bother you with things you don't want
- ✓ Be flexible to do almost anything you may want
- ✓ Help you to achieve your goal



# Other requirements

- ✓ Multiplatform
- ✓ Provide set of "tools" for various tasks and means to combine them to solve more complex problems
- ✓ Provide various user interfaces (CLI, GUI, web-based, automation)
- ✓ Can work with multiple databases simultaneously
- ✓ Allow easy data and metadata querying
- ✓ Allow use of results from data nad metadata query as input for other actions
- ✓ Extensible





# Python to the rescue...

- Multiplatform
- Open Source
- Strongly Typed, but Dynamic
- Easy to learn
- Powerful means to work with lists and dictionaries (and other data structures)
- Huge code repository for almost anything
- Libraries to build various user interfaces (CLI, GUI, web-based, automation) easily
- Embeddable in user application



# How it would look like?

Think about interactive console/shell that will give you all power of Python interpreter and will also understand SQL and your own commands, and you can tailor it to your specific needs.

It will have several incarnations...

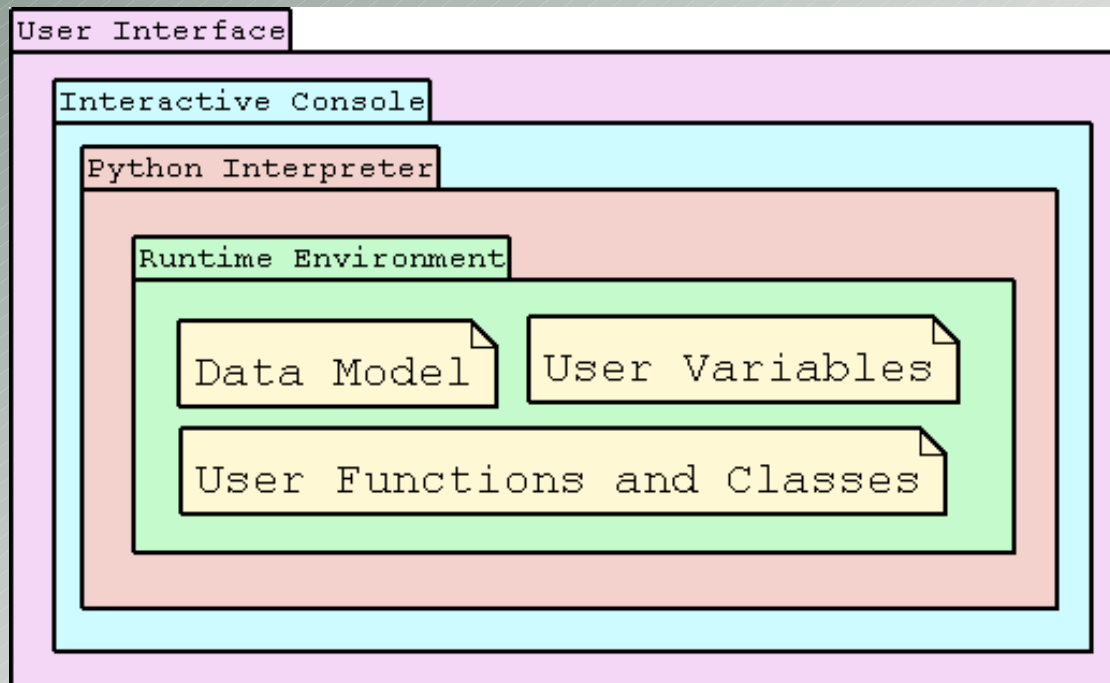
- ✓ CLI console (ISQL on steroids) to run scripts and work over SSH with remote servers.
- ✓ GUI console with tooltips, command completion, syntax highlight, option dialogs, shortcuts and wizards.
- ✓ Web application to work with databases hosted by your Webhosting Service Provider.





# Basic PowerTool™ design

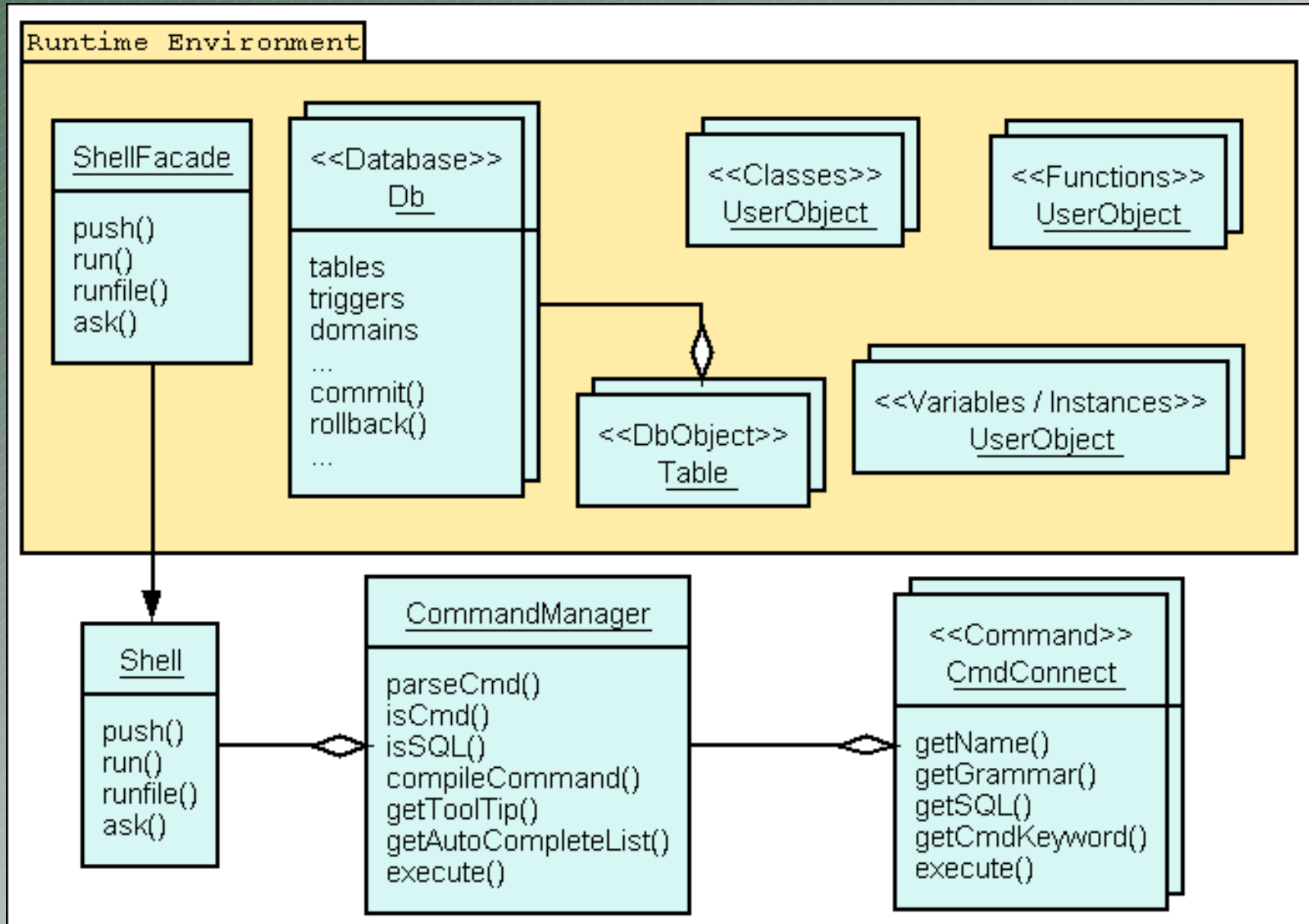
- Built using Python Interpreter
  - Wrapped into Interactive Console
    - Wrapped into UI



# Basic PowerTool™ components

- Set of Classes to work with Data Model
  - Metadata encapsulation into objects
  - Could be used independently
- CommandController and Commands Classes
  - Predefined and user-defined commands
  - Interacts with Runtime Environment (uses Data Model)
- UI Controller
  - CLI console, GUI, Web-based





# GUI Prototype In Action



**Do you like what you  
saw?  
So why not take it a  
little bit further?**





# Consider this...

- ✓ You'll have all power of Python and Python library at your disposal to solve (almost) any problem.
- ✓ You can wrap frequently used "solutions" into functions and classes for later use.
- ✓ You can write your own commands.
- ✓ You can modify and extend the UI to your needs.
- ✓ You can embed it at (various levels) into your own UI or application.
- ✓ You can share anything from the above with others.



# Plans for the future

- Start a project under Firebird umbrella
- Collect ideas
- Complete and Enrich the Data Model
- Commands for common task
- Finish the GUI front-end, create CLI and web-based one
- Add SQL documentation into help system
- Include it in standard Firebird distribution (at least on Windows)



**Questions ? Comments ?**



I thank you for your attention.

# The end

## Making You Own Firebird PowerTool with Python

Speaker: Pavel Císař

IBPhoenix

Czech Republic

