

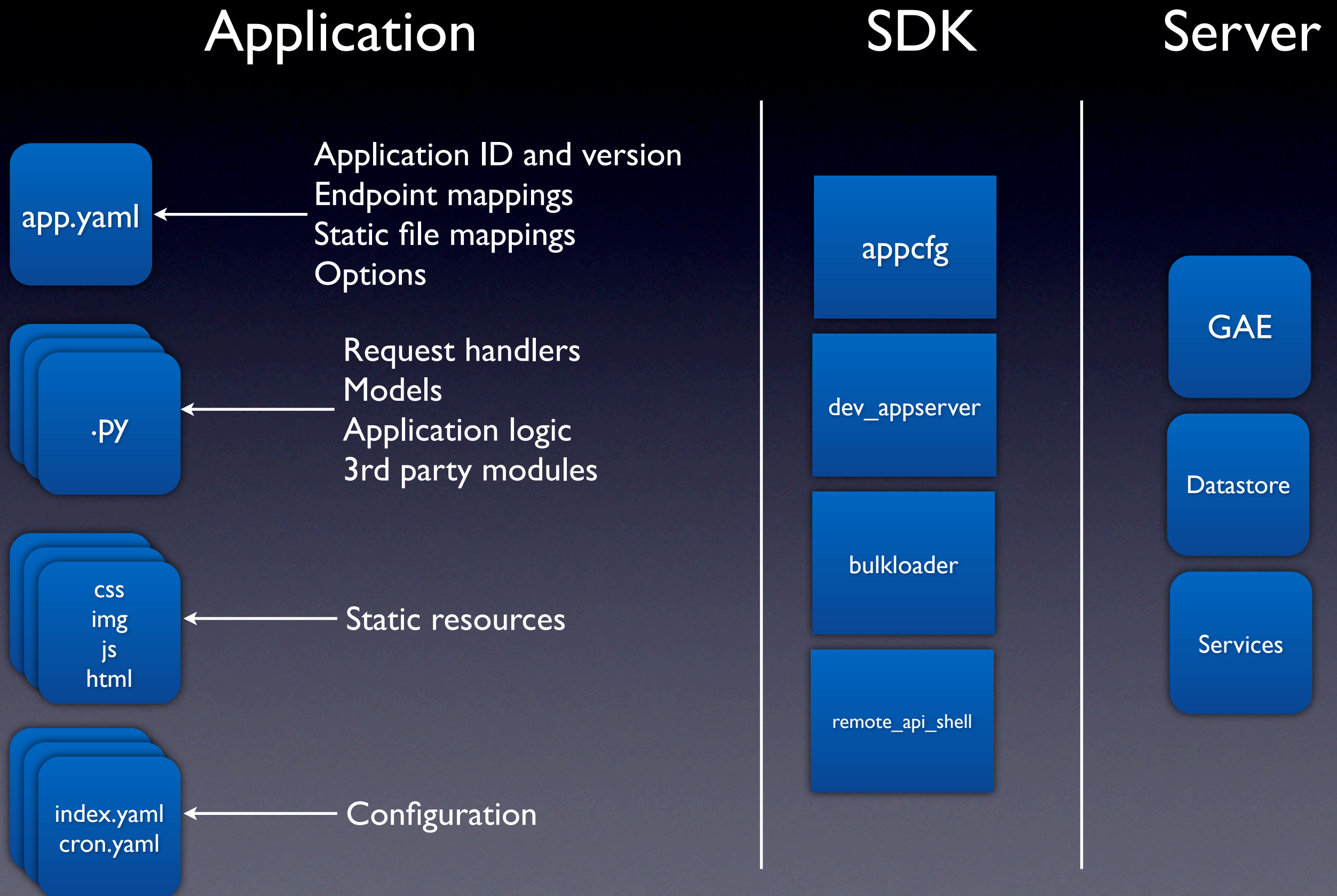
Google App Engine - Python

Douglas Starnes
SciPy 2011
Austin, Texas

Agenda

- ‘Lightning Round’ of GAE features
- Overview
- Data storage (datastore, blobstore, memcache)
- Task management (queues, backends)
- Development and Deployment Environments
- Odds and Ends
- Integration With SciPy

Overview of GAE Python



Datastore

- Non-relational structured table storage
- “NoSQL” & “column oriented”
- Based on Google’s BigTable

Datastore

- Define data model (kind) using objects
- Base db.Model class implements select, insert, update, delete
- Expando base class for dynamic properties
- First time object is inserted (put) datastore will handle the rest
- GQL queries for select only, all others programmatic access

Datastore

- Indexes for simple queries are built automatically
- Queries with multiple sort orders and multiple inequality comparisons need manual indexes (index.yaml)
- Cannot index blob or large text properties
- Support for transactions and asynchronous queries
- bulkloader & remote_api_shell
- Viewer and statistics in web admin

Blobstore

- Unstructured & binary data storage
- Blobs are immutable
- Unrelated to BlobProperty in datastore
- Can download blobs through web admin
- Maximum size == 2GB / Maximum read == 1MB
- Blobs uploaded via web forms*
- Experimental support for virtual file system to programmatically create blobs

Memcache

- In memory key/value storage
- Intended for short term storage
- Values expire or can be released if system is low on memory
- Maximum size of an entry == 1MB

Task Management

- GAE Python does not (currently) support multi-threading
- webapp limits requests to 30 seconds
- Task queues have a 10 minute limit
- Backends are GAE instances with no time limit (also have more memory and CPU time, more configurable)
- Backends cost extra!
- “Cron” jobs also supported (cron.yaml)

Enviroment

- Officially Python 2.5.2 is supported
- PyYAML and simplejson included
- Several modules (marshal, socket) are empty
- Any pure Python code (no C extensions) will work

SDK

- Windows, Mac OS X and Linux
- Command line and UI tools (UI not on Linux)
- Local development server
- Interactive console
- Services emulated on local server
- Open source

Deployment

- Command line or UI tool (no FTP)
- Versioning
- Code download
- Access to raw server logs
- Web-based admin

Odds and Ends

- Runtimes for Java and Go
- Authentication against Google accounts
- AppStats
- gdata-python-client
- Django-nonrel
- kay-framework (“Kay makes Appengine Yummy”)
- <http://code.google.com/appengine>

Integration With SciPy

- ProtoRPC
- Pull Queues

Questions?

douglas@poweredbyalt.net
www.poweredbyalt.net