So, I want to use MTS in my own projects. How do I get started?
MTS engine architecture

Deployment scenario
- Java host application
- HTTP web responder
- Shell: demo.py

Tailoring package
- MTS engine

Programming language
- Jython
- Python

Operating system
- Windows
- Mac
- Linux

Components that need installing

Deployment scenario
- Java host application
- HTTP web responder
- Shell: demo.py

Tailoring package
- MTS engine

Programming language
- Jython
- Python

Operating system
- Windows
- Mac
- Linux
About Jython

- Full implementation of the Python language, but in Java instead of in C
- Easy interaction with Java – Jython code can call Java libraries and (with some help) vice versa
- Lags behind CPython by a few versions (but closing the gap)
  - Sun recently hired the Jython project lead
- Similar Java-based implementations exist for Javascript (Rhino), Ruby (JRuby), and other languages; some languages, like Groovy and Scala, exist only on the JVM

Choosing Python or Jython

- Choose Jython if you're already a Java or JVM shop
  - We use Jython with the MTS Workbench application, which is written in Java
- Choose Python otherwise because it's faster, more widely used, and first to get new features and fixes.
  - Python with a web front end is our most common use case
Installing Python: necessary?

- Python is included with Mac OS X Leopard (10.5) and with many Linux distros.
  - Should be version 2.5 or later. Run `python` at the shell prompt to see the version.

- Need Python 2.5 -- must be installed separately under Mac OS X Tiger (10.4).

- Must be installed separately under Windows.

Python Installer

- Go to python.org/download and find the installer for your system:
  - Windows (XP or Vista)
  - Mac OS X (10.4 Tiger or 10.5 Leopard)

- Standard graphical installer
  - On Windows, there are a couple of post-install steps to allow you to run scripts at the command line; see python.org/doc/faq/windows or Google for “Python Windows FAQ”

- Source code provided for building under Linux, *BSD
Installing the MTS engine: Python

- Get the source code from http://chcr.umich.edu/mts
- At the command line, go to the source directory and run `python setup.py install`
- All the required libraries will be downloaded and installed as well

Installing Jython

- Go to jython.org and download the installer. For now, use version 2.2.1.
  - Version 2.5 will be a huge advance, but it’s not quite ready.
- Launch the jython installer jar file to launch a graphical installer. While not very pretty, it gets the job done. Default file locations are fine.
- Once installed, type `jython` at a terminal prompt to make sure it works.
Installing the MTS engine: Jython

- More complicated under Jython because the modern Python packaging tools do not run yet

- In your Jython folder, make a new folder called site-packages in lib/python2.2

- Download the tailoring engine source code from chcrumich.edu/mts

- Manually copy these items from the tailoring2 distro folder into site-packages:
  - tailoring2 subfolder [the one with all the .py files]
  - lib/jython-stdlib-additions/*.[py] [all the .py files]
  - lib/third-party/elementtree folder
  - lib/third-party/simplejson folder

Running the tests

- In the terminal, navigate to the tailoring2 source folder:
  - This is the one with 'doc', 'lib', and 'setup.py' in it

- Type `python tailoring2/test/test_all.py` to run the tests under Python
- Type `jython tailoring2/test/test_all.py` to run the tests under Jython

- If the tests don’t pass, the most likely cause is an issue installing the support libraries
What about IronPython?

- Like Jython, an alternate implementation of Python
  - Runs on .NET
  - Developed by Microsoft (and led by the original author of Jython)
  - More current than Jython (although Jython is catching up)
  - Ironpython.com

- Don’t know whether it works with the MTS engine. I wouldn’t anticipate major problems, but it hasn’t been tested. Great contribution opportunity!

- IronPython may need a couple support libraries to work with the tailoring engine.
  - 1.x: ElementTree and SimpleJSON
  - 2.x (beta): SimpleJSON